Deep Integration of Vision and Language

Benno Krojer
Doctor of Philosophy

School of Computer Science
McGill University
Montreal, Quebec, Canada

August 20, 2026

A thesis submitted to McGill University in partial
fulfillment of the requirements of the degree of
Doctor of Philosophy

Abstract

A major challenge in current AI systems is how to deeply integrate visual perception and language. Many of the tasks we humans face daily, and thus the tasks we want an AI to solve, involve an intricate mix of diverse signals — ranging from abstract text to various visual cues. For example, when assembling IKEA furniture with a friend, we combine instructions ("fasten part A to part B") and our friend's verbal suggestions with dozens of visual depictions of furniture parts, assembly steps as well as the actual physical pieces in front of us.

This thesis investigates whether current vision-and-language models genuinely integrate the two modalities or rely on shallow connections, approaching this question from three angles: behavioral evaluation, modeling, and interpretability: I introduce a diagnostic benchmark in which models must select the correct image from a set of minimally different video frames given a contextual description, testing whether they genuinely reason across both modalities or rely on shallow shortcuts. I extend this minimal pairs approach to image editing, curating training data from videos and simulation engines to cover action- and reasoning-centric edits largely absent from object-centric editing datasets. On the modeling side, I show that a model trained solely on image generation can be repurposed as a competitive vision-language understanding model, thereby bridging the two traditionally separate computer vision paradigms of generation and understanding. Finally, for interpretability, I study how LLMs represent visual tokens in their linguistic embedding space, introducing an interpretability tool that reveals far more visual tokens reside near interpretable language representations than prior methods suggest.

Together, these contributions advance our understanding of the deep inte-

gration of vision and language in current models, from behavioral evaluation and data to modeling and internal representations. As the field moves toward models trained from scratch on multimodal generation and reasoning beyond just text tokens, important questions remain: How do models internally accomplish all of this, e.g., their mechanisms for cross-modal reasoning and representing visual and linguistic concepts in the same space?

Abrégé

L'un des principaux défis des systèmes d'IA actuels consiste à intégrer étroitement la perception visuelle et le langage. Bon nombre de tâches auxquelles nous, les humains, sommes confrontés au quotidien, et donc celles que nous souhaitons voir résolues par une IA, impliquent un mélange complexe de signaux divers, allant du texte abstrait à divers indices visuels. Par exemple, lorsque nous montons des meubles IKEA avec un ami, nous combinons les instructions (« fixer la pièce A à la pièce B ») et les suggestions verbales de notre ami avec des dizaines de représentations visuelles des pièces du meuble, des étapes de montage ainsi que les pièces physiques réelles qui se trouvent devant nous.

Cette thèse examine si les modèles actuels de vision et de langage intègrent véritablement ces deux modalités ou s'appuient sur des connexions superficielles, en abordant cette question sous trois angles : l'évaluation comportementale, la modélisation et l'interprétabilité : Je présente un benchmark diagnostique dans lequel les modèles doivent sélectionner l'image correcte parmi un ensemble d'images vidéo présentant des différences minimes, à partir d'une description contextuelle, afin de vérifier s'ils raisonnent véritablement à travers les deux modalités ou s'ils s'appuient sur des raccourcis superficiels. J'étends cette approche des paires minimales à l'édition d'images, en sélectionnant des données d'entraînement issues de vidéos et de moteurs de simulation afin de couvrir les modifications centrées sur l'action et le raisonnement, largement absentes des ensembles de données d'édition centrés sur les objets. Du côté de la modélisation, je montre qu'un modèle entraîné uniquement sur la génération d'images peut être réutilisé comme un modèle compétitif de compréhension vision-langage, établissant ainsi un pont entre les deux paradigmes tradition-

nellement distincts de la vision par ordinateur que sont la génération et la compréhension. Enfin, en matière d'interprétabilité, j'étudie comment les LLM représentent les tokens visuels dans leur espace d'embedding linguistique, en introduisant un outil d'interprétabilité qui révèle que bien plus de tokens visuels se trouvent à proximité de représentations linguistiques interprétables que ne le suggèrent les méthodes antérieures.

Ensemble, ces contributions font progresser notre compréhension de l'intégration profonde de la vision et du langage dans les modèles actuels, depuis l'évaluation comportementale et les données jusqu'à la modélisation et les représentations internes. Alors que le domaine évolue vers des modèles entraînés à partir de zéro sur la génération multimodale et le raisonnement au-delà des simples tokens textuels, des questions importantes subsistent : comment les modèles accomplissent-ils tout cela en interne, par exemple, quels sont leurs mécanismes de raisonnement intermodal et de représentation des concepts visuels et linguistiques dans le même espace ?

Contributions to Original Knowledge

This doctoral thesis contributes original advances to the fields of vision-and-language evaluation, modeling, and interpretability, with the unifying question of whether AI models deeply integrate vision and language rather than learning shallow connections between the two modalities.

- **Multi-image guessing game benchmark (Chapter 2):** We introduce ImageCoDe, a curated image retrieval benchmark in which models must select the correct image from a set of minimally contrastive candidates based on a pragmatically-grounded description, with images sourced from nearby video frames and 21K descriptions crowdsourced via a two-player reference game. Evaluation reveals a stark gap between state-of-the-art models, our own designed models, and humans (22.0% vs. 90.8% on video subset), demonstrating that vision-and-language models at the time fail to deeply integrate visual context when language requires pragmatic inference.
- **Turning Diffusion Models into Vision-and-Language Reasoners (Chapter 4):** We introduce DiffusionITM, a method that transforms any diffusion model into an image-text matcher by interpreting the noise prediction error as a score of how similar the given image and caption are. On top of this zero-shot transformation, we also propose hard-negative finetuning. DiffusionITM enables a fair comparison to discriminative vision-and-language models such as CLIP on a curated collection of vision-and-language reasoning benchmarks. We reach comparable performance with CLIP variants,

thus serving as a valuable proof-of-concept for the text-to-image community that these models are implicitly more than generators.

- **Bridging the gap between object-centric and action-centric image editing (Chapter 3):** We introduce the AURORA dataset, a training corpus for instruction-guided image editing curated from videos and simulation engines to cover action- and reasoning-centric edits severely underrepresented in existing datasets, alongside AURORA-Bench, an expert-curated evaluation benchmark covering 8 diverse editing tasks. The resulting model significantly outperforms prior editing models on action- and reasoning-centric tasks as judged by human raters, and we identify critical flaws in existing automatic metrics for semantically complex edits. Overall, we find that physical and reasoning edits are significantly more difficult than object-centric edits.

- **Revealing Highly Interpretable Visual Tokens in LLMs (Chapter 5):** We introduce LatentLens, an interpretability method that compares visual token representations to contextual LLM embeddings across layers, advancing previous methods such as Logit Lens to multimodal settings. Contrary to prior work, LatentLens reveals that the majority of visual tokens have interpretable nearest neighbors in the LLM embedding space throughout all LLM layers, providing evidence that LLMs represent visual information in their linguistic embedding space more systematically than previously understood.

Collectively, these contributions advance our understanding of vision-and-language integration in AI models: from rigorous diagnostic evaluation, through data-driven modeling that unifies generation and understanding, to interpretability tools that illuminate how visual and linguistic representations relate inside modern multimodal models.

Contributions of Authors

Chapter 2 (ImageCoDe) is based on Krojer et al. (2022), published at the Annual Meeting of the Association for Computational Linguistics (ACL), 2022. The initial direction was jointly developed between my supervisor, Siva Reddy, and me. Afterwards, I led all parts of the project from data collection and model training to writing the paper. Vaibhav Adlakha helped with setting up the data collection software, while Edoardo Ponti and Siva Reddy were directly involved in shaping certain sections such as the introduction. All authors participated in weekly project discussions and helped refine the paper.

Chapter 4 (DiffusionITM) is based on Krojer et al. (2023a), published at the Conference on Neural Information Processing Systems (NeurIPS), 2023. The initial direction stemmed from discussions between my supervisor Siva Reddy and me about recently released DALL-E and Stable Diffusion models. I led all parts of the project, from pinning down the exact research question and methodology to running experiments and curating the data. Elinor Poole-Dayan contributed the bias-related parts of the paper (assembling the bias dataset, running experiments on them, and writing bias-related paragraphs), as well as refining figures. Christopher Pal and Vikram Voleti helped during writing (especially the mathematical formulation and diffusion literature) as well as during rebuttal. All authors participated in project discussions and helped refine the paper.

Chapter 3 (AURORA) is based on Krojer et al. (2024a), published at the Conference on Neural Information Processing Systems (NeurIPS Datasets and Benchmarks Track, Spotlight), 2024. The initial direction came from discussions be-

tween myself and Christopher Pal. I led all parts of the project, from defining the exact problem and collecting the data to running all main experiments and writing the paper. Luis Lara assisted the data curation process, specifically refining the simulation engine and assets. Dheeraj Vattikonda helped with a secondary question around model internals and their attention maps. Finally, Eva Portelance helped with the introduction and background section. All authors participated in project discussions and helped refine the paper.

Chapter 5 (LatentLens) is based on Krojer et al. (2026), published at the International Conference on Machine Learning (ICML), 2026. I led all parts of the project from setting the initial direction and running most experiments to writing the published work. Shravan Nayak contributed ablations on downstream performance that are currently in the appendix. Marius Mosbach, Desmond Elliott and Siva Reddy were regularly involved in shaping the exact contributions of LatentLens. Marius Mosbach contributed major aspects of the background section, while Desmond Elliott refined introduction and Figure 3. Oscar Mañas designed Figure 1 and Figure 2. All authors participated in project discussions and helped refine the paper.

Acknowledgements

It is hard to put into words how meaningful these five years have been. This time in Montreal, Mila and McGill were, without a doubt, the best years of my life (so far). I am deeply grateful for the friendships that will last for a life, for having great mentors and later in the PhD also being a mentor, and for the ups and downs. I am grateful for being able to follow my curiosity and ideas freely.

Siva, thank you for the trust in me, even when I was doubting it all. For encouraging me to be ambitious. You are a big factor why I will continue in academia. I can only hope to ever cultivate a lab with a similarly immaculate vibes as ours. With that, I also want to thank all my lab members and collaborators over the years. This lab is truly special and it takes intention to keep it that way. Daily lunches, drinks and shenanigans do not happen by themselves.

I could not have imagined how much of a community I would find here over the years. The four years spent at the "Mila house" on Esplanade Avenue felt like a real home and I feel a deep bond to all the roommates, upstairs neighbors and "friends of the house": Thank you Darshan for being there for me as a roommate from my very first to my very last day in this city, for being the most accurate personalized recommendation system for TV shows (Ted Lasso, Avatar), for inventing the ball game and for being funny. Thank you to Vaibhav, for saying yes to any fun adventure (who else would spend 2 (!) nights in Scranton with me), for setting the foundation of what would become the long Esplanade tradition, and for our friendship. Thank you Nathan, for moving out so we could move in — and, more importantly, for being someone I could share the vulnerable with, for having such a similar mind that I knew I made a close friend in my first

week, and for finding spiritual enlightenment together. Thank you Melisande for documenting our lives in comics, Mizu for forming the legendary Darshan-Mizu-Benno trio, Shashank for always being the most chill and kind guy, Tom for being an awesome podcast co-host (and much more), David (French) for inspiring me every day to have the worst ideas. And so many more Esplanaders that made this time so special: David (the Paler), Xiusi, Ethan, Rasha, Ching Lam, Adriana, Carling, ... thank you!

I am grateful for the life outside of academia that kept me sane (and fit). The McGill Ultimate Frisbee community is full of love and, as a grad student among undergrads, let me stay young via the latest memes and slang. Special thanks to the Montrey'all crew in Amsterdam, to Jack for the "dogged determination to remain a child" [1], and to Solomon, Nate, Georges and Hayden (with his penguin) for cherished friendships and hard-fought tournaments.

While having wonderful friends and hobbies is important during a PhD, I also really enjoyed the actual process of doing science: the unique culture at Mila, how we share ideas, the collaborations and countless chats in the kitchen, and community service in various forms. I will be forever grateful for early encouragement during undergrad to be ambitious and just try things (Denis Peskoff, Nora Kassner, Niklas Koehler, Julian Jungwirth). Later on in the PhD, I was deeply inspired by our postdocs at the time: Marius Mosbach, Verna Dankers, Karolina Stańczak. They left me more certain than ever that staying in academia is the right path. Thank you Marius for helping me finish the PhD on a high note, bringing the German out in me and making our lab mega social. Running the Mila-wide "tea talks" and a reading group for most of my PhD taught me a lot about how sharing and doing science is far beyond the formal publication process: lively debates during talks, discussing papers, sharing half-baked project pitches. Thank you Oscar Manas for running the Language Grounding reading group for many years with me (and for being a friend for life). Likewise, Rabiul Awal, I admire your conviction to pursuing hard research

[1] A quote from our Sports day video (Link) where we do 26 sports (A to Z) in one day.

problems and your never-ending curiosity and excitement. I am grateful for my fellow tea talk organizers, keeping the show running: Rim Assouel (and for our matcha & cog-sci mutual interests), Lena Néhale Ezzine, Tom Marty, Divyat Mahajan, Eric Elmoznino, Zihan Wang. Thank you Mandana for being a cornerstone of the Mila community and for the runs, bike rides, matchas and deep chats. Countless people have shaped how I think about research over the years. Thank you for making me into the researcher I am today. A special thank you goes to my mentors, next to Siva himself: Chris Pal, Mido Assran, Aishwarya Agrawal, Desmond Elliott and the aforementioned postdocs (including Edoardo Ponti & Eva Portelance). Life was never boring, and full of support, thanks to Gaurav Kamath, Cesare SDP, Christos Tsirigotis, Xing Han Lu, Sara Marjanovic, Michelle Lin, Samuel Lavoie, Alexey Ostapenko, Sonia Joseph, Michael RM, Dora Jambor, Sophie Xhonneux, Embo (R.I.P.), and many more.

Elinor, I could not have done this without you. You met me during a low time when I was questioning the whole PhD, and here we are now, starting a new life in Boston as an academic power couple. I admire the kindness, love and caring you devote to the people around you, and I am grateful to be one of them. You always make me see the fun and whimsical side when I got too worried about some deadline or the state of the world. Thank you.

Finally, thank you mom and dad for everything. You always inspired me to chase my dreams, to take risks and the value of doing the right thing, of being kind to everyone, of asking why, of curiosity, and of doing science not for the h-index but to actually advance and preserve knowledge for humanity.

I am excited for the journey ahead as a postdoc. I have learned from the best postdocs what it means to raise a new generation of scientists, and to cultivate a lab culture of support. I can only hope to come anywhere close to that.

P.S.: Acknowledgements are arguably the most interesting part of a thesis.[2] It was impossible to put 5 years into these 3 pages. So for more details, please read "Behind the Scenes" in Chapter A (a section also found in my papers).

[2] I recommend The unexpected poetry of PhD acknowledgements to the reader.

Contents

List of Figures

List of Tables

Part I

Introduction

1

Introduction

1.1 Motivation

When we think of human-like Artificial Intelligence (AI) models, we most commonly picture a system handling either language *or* vision input. Designing models that behave intelligently on a single modality alone is already a major undertaking. However, when one tries to develop a model with the ability to seamlessly process and reason about *both together*, countless new challenges and scientific questions arise. For example, should we aim for one monolith architecture in which both modalities are unified (Team, 2024), or have specialized modules for text and images separately (Liu et al., 2023)? When do models rely too much on one of the modalities while ignoring the other (Goyal et al., 2017b)? And how is a given concept represented inside models, either as part of a visual, linguistic or multimodal latent space (Goh et al., 2021; Krojer et al., 2026)?

It is not a coincidence that the field of AI has focused on vision and language

as the two canonical modalities. After all, perception and language are both foundational and complementary for humans to understand the world around us (Bisk et al., 2020a). In this thesis, I study the rich connections between the two modalities, asking: **Is an AI model deeply integrating vision and language in its representations and behavior, or has it merely learned a shallow connection?**

Human lives are inherently multimodal, and thus also the tasks we care about when evaluating AI systems. We reason about and make sense of complex situations that involve abstract signals, often in the form of language, combined with perceptual observations. Likewise, we expect a physical AI system like a robot not only to perceive and interact with objects, but also to follow human instructions and draw from internet knowledge to solve a task (ichter et al., 2023). Moreover, we expect a digital agent like a web assistant not only to read the website's source code, but also to navigate and interact with the visual layout of the website and various media such as images and videos (Koh et al., 2024).

Perception and language cover complementary aspects of understanding: Perceptual understanding spans from recognizing static scenes of objects and attributes (Krishna et al., 2017) to tracking dynamic events across time (Zellers et al., 2021), and extends to high-level reasoning about intuitive physics (Riochet et al., 2022; Spelke and Kinzler, 2007) such as object permanence, affordances, or cause-and-effect. On the other hand, language is a more abstract medium of communication (Tomasello, 2010). It enables both humans and AI systems to efficiently share what we see, hear, and interact with. In particular, language can act as a symbolic layer over perception, e.g., through taxonomies (Rosch et al., 1976), metaphors (Lakoff and Johnson, 1980), or explanations (Lombrozo, 2006). Crucially, the meaning of language is not fixed by semantics alone (Heim and Kratzer, 1998): pragmatics (Clark, 1996; Bisk et al., 2020a) shows that meaning can often depend on a shared perceptual context, thus making language inseparable from the perceptual grounding it abstracts over.

With such rich and complementary signals from both modalities, one central challenge of the field of vision-and-language is to **design AI systems that deeply**

integrate the two modalities into a single model that exhibits flexible cross-modal reasoning behavior (Tong et al., 2026; Ray et al., 2026). For example, one desideratum of such a model would be that certain components (e.g., certain layers) would reach a modality-agnostic abstract understanding of the world (Nikankin et al., 2025; Wu et al., 2025; Huh et al., 2024; Devillers et al., 2024), akin to the semantic hub hypothesis (Patterson et al., 2007) in neuroscience.[1] The strongest realization of this goal is the recent paradigm of *unified* (or native multimodal) models, usually trained jointly from scratch on both modalities with no unimodal pretraining stage (Team, 2024; Tong et al., 2026).

Throughout this thesis, I use *deep integration* in the sense of the following working definition.

Deep Integration of Vision and Language

A model *deeply integrates* vision and language to the extent that the two modalities interact and are shared within it, rather than forming two loosely connected unimodal systems. This can be characterized along several axes (Figure 1.1):

- **Representational:** vision and language occupy a shared, modality-agnostic space rather than separate subspaces.
- **Architectural and training:** the modalities attend back and forth, share weights, and are trained jointly rather than stitched together after separate pretraining.
- **Behavioral:** the model flexibly reasons back and forth across the modalities, and exhibits this on certain benchmarks.

Humans achieve precisely this kind of deep integration, fluidly combining vision and language to solve various tasks. The premise of my thesis is that while we want to imbue AI models with this integration too, it is non-trivial to know

[1] The hypothesis posits that, while different modalities (vision, audio, touch, etc.) have associated brain regions, they all form a holistic semantic representation in an amodal region of the brain.

whether it is present, and if it is, how models implement it. I therefore study this challenge from three angles, each probing one of the axes above: 1) *behavioral evaluation* (the behavioral axis); 2) *modeling innovations* (the architectural and training axis); and 3) *interpretability* (internal evaluation of the representational axis). Before turning to these contributions, I sketch a brief history of how the field has approached modality integration, from early visual BERT variants to recent unified multimodal models.

A short history of integration paradigms. Combining vision and language is non-trivial, each one with fundamentally different properties and structure: Vision is input to the model as continuous 2-D or 3-D tokens with no well-defined atomic units. Language, by contrast, is discrete and symbolic with clean human-defined units (words, sentences) and structure (grammar). For much of their history, Computer Vision (CV) and Natural Language Processing (NLP) were mostly separate fields, and bridging them has been far from straightforward (Figure 1.1).

A first mainstream convergence of the field arrived with large-scale image-caption datasets (Lin et al., 2014; Krishna et al., 2017; Sharma et al., 2018) and the development of BERT (Devlin et al., 2019), giving rise to Vision-Language (VL) BERT variants (Lu et al., 2019; Chen et al., 2020a; Tan and Bansal, 2019). Unlike their unimodal predecessors, VL BERTs let both visual and language tokens attend to each other across many attention layers (Bugliarello et al., 2021), making the cross-modal interaction genuinely bidirectional: text could draw on visual context, and vision on linguistic context. Yet the widely adopted CLIP-paradigm highlights that this deep fusion can come at a cost: In order to efficiently apply contrastive learning on internet-scale data, (Radford et al., 2021a) trades many layers of cross-attention for a single dot product instead. Soon after, the unprecedented capabilities of LLMs not only changed the field of NLP (Brown et al., 2020; Touvron et al., 2023), but also meant it was more practical to extend LLMs with vision instead of redesigning architectures from scratch as multimodal (Alayrac et al., 2022; Liu et al., 2023). These multimodal LLMs connect

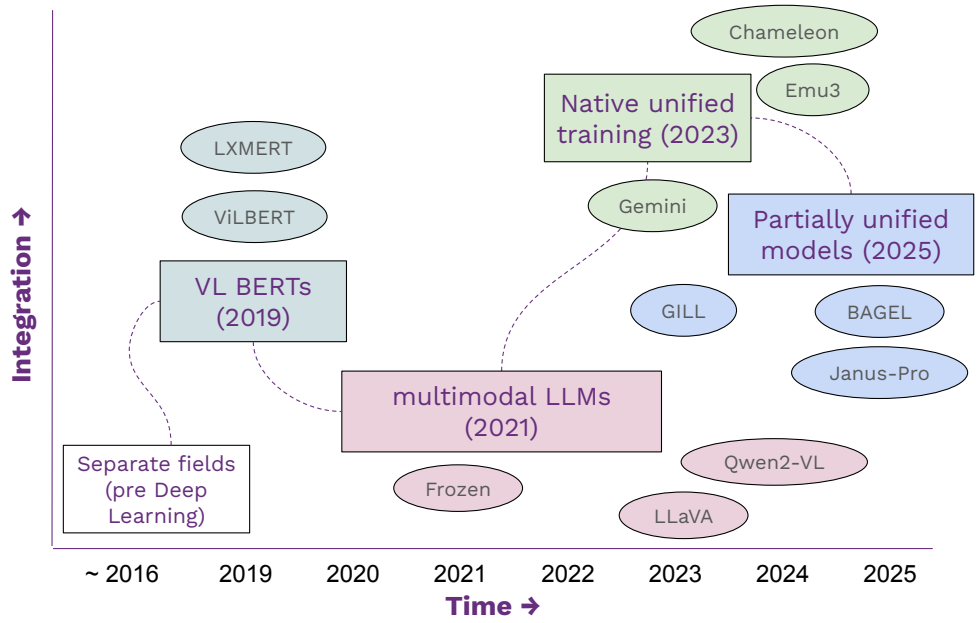


Figure 1.1: A schematic history of modality integration in vision-and-language models, showing to what extent modeling paradigms deeply integrate vision and language, considering: a) back-and-forth interactions (e.g., cross-attention layers), b) joint (pre)-training, c) shared weight space and d) shared objective and capabilities (e.g., autoregressive generation). We can observe that integration is not simply monotonically increasing. Note: certain key paradigms such as CLIP (Radford et al., 2021a) and diffusion-based text-to-image models (Rombach et al., 2022) are omitted. If included, CLIP would rank lowest while diffusion models would rank similarly to multimodal LLMs.

a pre-trained visual encoder to an LLM via a small learned connector (Li et al., 2023b, 2024a; Liu et al., 2023), at the cost of losing the "attention symmetry" of the visual BERT era. For modality integration, this is a step back: only text tokens can attend to vision, but vision is not informed by text, and the bulk of training effort is invested in the unimodal backbones rather than in cross-modal integration. The field has since moved to question this modular approach, exploring full joint training on multimodal data rather than connecting largely frozen unimodal backbones (Tong et al., 2026; Team et al., 2025; Wang et al.,

2024c). Chameleon (Team, 2024), an early pioneering work in this direction of *unified models*, treats both text and image as tokens in a single auto-regressive sequence, trained jointly from scratch. However, many subsequent models do not commit to full unification. They differ from strict unification in training setup (often initializing from pretrained LLM backbones), in objective design (using separate objectives for visual generation), and in weight sharing (a large proportion of weights is not shared across modalities) (Deng et al., 2025; Tong et al., 2026; Serra et al., 2025). The big picture throughout all these paradigms is that the field has not converged on how deeply vision and language should be integrated, and whether the paradigm of "full unification" is what we should aim for.

1.2 Guiding Challenges and Questions for the Thesis

Throughout the historic developments of vision-and-language integration outlined above, a few key questions and challenges remain constant. This thesis focuses on the following three:

The first challenge is to design evaluations that tell us whether vision and language are genuinely integrated in multimodal models. Across any domain in AI, models are prone to take shortcuts and flawed benchmarks can fail to detect this (Geirhos et al., 2020; McCoy et al., 2019). Thus scores which should ideally indicate genuine vision-and-language understanding can often become instead misleading and inflated. Especially in vision-and-language tasks, this is a long-standing problem since models tend to fall back to language priors while ignoring the visual input (Goyal et al., 2017b; Tong et al., 2024; Hsieh et al., 2024). Chapter 2 (Krojer et al., 2022) of the thesis addresses this question: How can we design complex tasks and datasets that avoid shortcut issues and instead incentivize deep back-and-forth reasoning between perception and language (i.e., modality integration)? One fruitful evaluation methodology for measuring genuine understanding is *minimal pairs* of inputs that vary in small meaningful aspects: if we vary only one part of the input, will the model change

its prediction accordingly? This idea was pioneered in language-only settings with tasks such as the Winograd Schema Challenge (Levesque et al., 2012), where minimal pairs are designed to test world knowledge via resolving pronoun ambiguity. As it became clear that vision-and-language models in particular are prone to relying on superficial textual cues rather than reasoning over the visual context, a myriad of minimal pair datasets emerged (Thrush et al., 2022b; Parcalabescu et al., 2022; Yuksekgonul et al., 2023a; Hsieh et al., 2024), either for evaluation or training. While most studies curate minimal *textual* pairs (Parcalabescu et al., 2022; Hsieh et al., 2024; Yuksekgonul et al., 2023a), these pairs can still be prone to shortcuts such as "blind" baselines performing above chance (Hsieh et al., 2024). Thus, in ImageCoDe (Krojer et al., 2022) I focus on *visual* minimal pairs: while harder to curate, they fully avoid text-only shortcuts and test unique vision-centric phenomena. Specifically, Chapter 2 identifies videos as a natural and rich source of minimal changes and curates nearby video frames as minimal pairs. As I will also show in Chapter 2, the way ImageCoDe defines minimal change as the smallest visual change that a human can still put into words leads to a level of minimal visual differences unmatched by other related datasets (Thrush et al., 2022b; Awal et al., 2024). Beyond the design choices for constructing such minimal visual pairs, Chapter 2 also studies concrete vision-and-language phenomena central to multimodal reasoning such as spatial reasoning, language-guided zooming and pragmatics. In line with the thesis theme of deep modality integration, we also study the effect of increasingly deeper integration of visual context into the model processing, such as allowing the model to attend to several distractor images at once. Overall, ImageCoDe remains a challenging test bed even for current vision-and-language models (see Chapter 6), providing an evaluation that is difficult to saturate via superficial shortcuts.

A second major challenge is to unify visual generation and understanding in a single vision-and-language model. The above ImageCoDe task was designed as purely discriminative: given 10 highly similar images and a de-

scription, select the correct image. In Chapter 3, we target the inverse: Instead of discriminating minimal changes between images, can we train a model to *generate* such changes based on a natural language prompt? To solve this task of text-conditioned image editing (Brooks et al., 2023), we need to *integrate* visual understanding (i.e., how the given image relates to the text prompt) and visual generation (i.e., generating the requested change) in a single model. This setup tightly couples a more language-centric skill involving semantics (visual understanding) with a more vision-centric one involving fine-grained visual details (visual generation). Curating training data for editing in the form of image pairs with a single meaningful change is hard (Zhang et al., 2024) and automated pipelines often introduce noisy pairs with too many changes (Brooks et al., 2023). Moreover, many studies on image editing focus on simpler static edits that add or remove objects and change their attributes. In Chapter 3 we instead focus on edits that require more reasoning about the dynamics of actions and spatial movement. Similar to ImageCoDe, we draw from videos to curate high-quality image pairs that depict only a *single* change, together with human annotations of that minimal change. AURORA is released as a complete package: training data, a fine-tuned model, and an evaluation benchmark.

Beyond better evaluation and training data, achieving deeper modality integration also requires modeling innovations. One long-standing hypothesis is that unifying visual *generation* and *understanding* in a single model could lead to more integrated representations and reasoning (Yu et al., 2024; Zheng et al., 2026). Intuitively, generating an image from scratch should discourage shortcut learning more than simply classifying or ranking one (Geirhos et al., 2020): to generate a correct scene, a model may need to reason more carefully about how objects, attributes and the prompt compositionally relate to each other (Andreas et al., 2016), rather than exploiting surface correlations. In Chapter 4 (Krojer et al., 2023a), we ask whether a generative model like Stable Diffusion (Rombach et al., 2022), explicitly trained only on denoising images, can implicitly also act as an understanding model. With DiffusionITM in Chapter 4, we provide evidence

that this unification is indeed feasible, re-framing Stable Diffusion to provide a score to image-text pairs. Re-framing Stable Diffusion to score image-text pairs turns out to be a meaningful test of understanding: several established vision-language reasoning benchmarks are naturally cast as image-text matching tasks (Thrush et al., 2022b; Yuksekgonul et al., 2023a). However, our study also finds that better generation does not always seem to imply better understanding: newer and stronger generative models do not consistently outperform discriminative ones on several reasoning benchmarks, echoing a broader philosophical question: What is the relation between creation and passive understanding (e.g. discriminating given options)? In humans we intuitively expect creation to imply an understanding at least as sophisticated as passive understanding of the content alone (e.g., being able to answer questions about it), exemplified by Richard Feynman's famous quote "What I cannot create, I do not understand". For example, someone who has written a mathematical proof would have a deeper understanding of it than someone who can merely follow the proof and answer questions about it. However in AI it is an open question whether models can generate content without understanding what they created (dubbed the *Generative AI Paradox*; West et al., 2024).

Finally, the third challenge is interpreting how visual and linguistic representations relate to each other inside models. The above behavioral studies, even if done rigorously via diagnostic minimal pair setups, have their limits. No matter how carefully we design models with multimodal unification in mind (architecture, objectives, or data), it is a separate question whether they *actually* end up as unified internally: even a model trained in the most unified way possible may still learn to process and represent vision and language separately (Serra et al., 2025). What does it then mean for a vision-and-language model to be unified? Chapter 5 contributes to this question by studying the puzzling situation of why it is so easy to connect a vision encoder to a frozen LLM with minimal adaptation — and show how visual tokens are regularly mapped close to interpretable LLM representations. Early work establishes that connecting

fully *frozen* vision and language models works well, even with very simple mapping networks (Merullo et al., 2023; Lu et al., 2022), and that LLMs implicitly learn visual priors during text-only pre-training (Han et al., 2025). The Platonic Representation Hypothesis (Huh et al., 2024) would imply that such alignment should be easy as it posits that models trained separately on vision and language respectively may nonetheless converge to similar representations of the underlying world. What is still missing, however, is a detailed account of how LLMs represent visual tokens internally. If a frozen LLM trained only on language can process visual tokens (treating them from the input layer the same way it would text tokens) then those tokens should in some sense resemble language from the very start. For example: are visual tokens mapped to interpretable text embeddings of the LLM?

Existing work paints a mostly pessimistic picture: modalities occupy largely separate activation subspaces, with visual tokens forming their own narrow cone far from language representations (Shukor and Cord, 2024). Moreover, visual tokens are not known to have interpretable nearest neighbors in the LLM embedding space (Mokady et al., 2021; Neo et al., 2025), and only in later layers does interpretation via the LLM unembedding matrix become feasible at all (Neo et al., 2025). Chapter 5 introduces a new interpretability tool (LatentLens) showing that past work underestimates how *interpretable* visual tokens are, i.e., how many of them reside close to semantically meaningful LLM representations. Our key insight is that *contextual* LLM embeddings are the right source of potential nearest neighbors. We design a control setup of finetuning only a simple MLP connector mapping visual tokens into the embedding space of a frozen LLM (Deitke et al., 2024). Our experiments show that the majority of visual tokens are judged as interpretable even at the input layer across a total of fifteen VLMs (our controlled setup & off-the-shelf models). Broadly, LatentLens is a convincing case study for why interpretability tools, often developed with LLMs in mind (nostalgebraist, 2020; Elhage et al., 2021), may need to be tailored to multimodal settings.

1.3 Thesis outline

This thesis is split into three parts: diagnostic evaluation (Chapter 2), improving models (Chapters 3 to 4), and interpretability (Chapter 5).

Chapter 2: Image Retrieval from Contextual Descriptions. Chapter 2 introduces ImageCoDe, a benchmark for fine-grained multi-image reasoning that combines 21K crowdsourced pragmatically-grounded descriptions with sets of highly similar video frames. To elicit diverse phenomena, we asked pairs of annotators to play a reference game over near-identical images, naturally requiring visual search, grounded pragmatics, and multi-image reasoning. We evaluate a range of models, from CLIP to architectures with deeper cross-modal fusion such as ViLBERT and UNITER, and also develop model components for multi-image context integration. My retrospective reflection in Chapter 6 shows that years later ImageCoDe has still not saturated as a task.

Chapter 3: Learning Action- and Reasoning-Centric Image Editing. Chapter 3 studies text-guided image editing, a task that unifies understanding and generation: a model must first understand the scene before it can generate the requested change. It is also the inverse of the minimal pair paradigm: rather than asking whether a model can discriminate between minimally different images, we ask whether it can generate one. We introduce the AURORA dataset, curated from videos and simulation engines to cover action- and reasoning-centric edits largely absent from existing object-centric datasets for editing. The result was the strongest physical-action-based editing model at the time of publication.

Chapter 4: Diffusion Models as Zero-Shot Image-Text Matchers. Chapter 4 asks whether a generative model like Stable Diffusion can act as a vision-and-language understanding model at the same time. We show the feasibility of this empirically: reframing Stable Diffusion to score image-text pairs on compositional reasoning benchmarks unifies competitive image generation *and* understanding in a single model. We also address an "asymmetry" between text retrieval and image retrieval settings via normalization of the score, and propose a tailored fine-tuning approach.

Chapter 5: Revealing Highly Interpretable Visual Tokens in LLMs. After having measured and improved various vision-and-language abilities, the question remains: how exactly do models internally accomplish this? Chapter 5 asks how current multimodal LLMs can process images with minimal additional training, even when almost all components are kept frozen and only a small connecting MLP is trained. We introduce LatentLens, an interpretability tool that compares visual token representations to contextual LLM embeddings across LLM layers, and find that visual tokens are far more interpretable than prior work suggested: the majority of visual tokens have interpretable nearest neighbors in the language space, throughout all LLM layers.

Chapter 6: Discussion and Future Directions. Chapter 6 reflects on the contributions of each chapter from the vantage point of 2026, discussing how each work has been picked up, what subsequent trends it connects to, and what looks different in hindsight. I then outline future directions, including progress toward truly unified multimodal models and extending chain-of-thought reasoning from text to visual tokens. Across these modeling directions, I discuss how interpretability can provide insights into how visual and linguistic knowledge interact and how truly multimodal reasoning might unfold internally.

1.4 Publications

The research presented in this thesis is based on the following peer-reviewed publications. Chapters are indicated in brackets.

- **Benno Krojer**, Vaibhav Adlakha, Vibhav Vineet, Yash Goyal, Edoardo Ponti, Siva Reddy. *Image Retrieval from Contextual Descriptions.* Annual Meeting of the Association for Computational Linguistics (ACL), 2022. [Chapter 2]

- **Benno Krojer**, Dheeraj Vattikonda, Luis Lara, Varun Jampani, Eva Portelance, Christopher Pal, Siva Reddy. *Learning Action and Reasoning-Centric Image Editing from Videos and Simulation.* Conference on Neural Information Processing Systems (NeurIPS, Spotlight), 2024. [Chapter 3]

- **Benno Krojer**, Elinor Poole-Dayan, Vikram Voleti, Christopher Pal, Siva Reddy. *Are Diffusion Models Vision-And-Language Reasoners?* Conference on Neural Information Processing Systems (NeurIPS), 2023. [Chapter 4]

- **Benno Krojer**, Shravan Nayak, Oscar Mañas, Vaibhav Adlakha, Desmond Elliott, Siva Reddy, Marius Mosbach. *LatentLens: Revealing Highly Interpretable Visual Tokens in LLMs.* International Conference on Machine Learning (ICML), 2026. [Chapter 5]

Beyond the content of this thesis, I have published the following first-author work.

- **Benno Krojer**, Mojtaba Komeili, Candace Ross, Quentin Garrido, Koustuv Sinha, Nicolas Ballas, Mido Assran. *A Shortcut-aware Video-QA Benchmark for Physical Understanding via Minimal Video Pairs.* Transactions on Machine Learning Research (TMLR), 2025.

Finally, I significantly contributed to the following works as a co-author:

- Ada Defne Tür, Gaurav Kamath, Joyce Chai, Siva Reddy, **Benno Krojer**. *Would you still call this Dax? Novel Visual References in VLMs and Humans.* Under Review, 2026.

- Sara Vera Marjanović, Arkil Patel, Vaibhav Adlakha, Milad Aghajohari, Parishad BehnamGhader, Mehar Bhatia, Aditi Khandelwal, Austin Kraft, **Benno Krojer**, Xing Han Lù, Nicholas Meade, Dongchan Shin, Amirhossein Kazemnejad, Gaurav Kamath, Marius Mosbach, Karolina Stańczak, Siva Reddy. *DeepSeek-R1 Thoughtology: Let's think about LLM Reasoning.* Transactions of Machine Learning (TMLR), 2025.

- Saba Ahmadi, Rabiul Awal, Ankur Sikarwar, Amirhossein Kazemnejad, Ge Ya Luo, Juan A. Rodriguez, Sai Rajeswar, Siva Reddy, Christopher Pal, **Benno Krojer**, Aishwarya Agrawal. *The Promise of RL for Autoregressive Image Editing.* Conference on Neural Information Processing Systems (NeurIPS), 2025.

- Oscar Mañas, **Benno Krojer**, Aishwarya Agrawal. *Improving Automatic VQA Evaluation Using Large Language Models*. AAAI Conference on Artificial Intelligence (AAAI), 2024.

Part II

Manuscripts

2

Image Retrieval from Contextual Descriptions

Preface

In Chapter 1, I broadly established that combining perception and language, e.g., for complex reasoning tasks, is far from trivial. Concretely, it requires compositionality on both modalities, decomposing a visual scene into its parts and the provided language into its sub-meanings; resolving referring expressions to zoom into certain parts of the visual context; and broadly a shared semantic representation space, among others. The contribution of this chapter is measuring such abilities in vision-and-language models at the time of publication.[1] This is the first of the thesis's three angles on deep integration (Chapter 1): a purely *behavioral* test of whether a model integrates the two modalities, before later

[1] A secondary contribution is on the modeling and analysis side.

chapters turn to modeling and internal representations.

What desiderata should we set for a general and hard vision-and-language benchmark? (1) It should cover a wide range of challenging phenomena at the intersection of vision and language beyond simple scenes and questions about single static images. For example, to solve many of the examples a model should have to reason back-and-forth between complex visual input and text descriptions. (2) The benchmark should not allow models to latch onto spurious correlations, e.g. text-only shortcuts. Instead, only models that deeply integrate vision and language in their representations and reasoning should do well. (3) It should be hard enough to track progress for several generations of future models without saturation.

The ImageCoDe task (Krojer et al., 2022), published at ACL 2022, covers these desiderata in the context of a two-player communication game over a set of ten minimally different images. To collect the data, we first identify sets of similar images, then ask one annotator to describe a target image such that a second annotator can reliably select it from the set. The final dataset comprises 21K descriptions with more than 90K images. Our setup is closely related to the visual reference games studied in grounded pragmatics (Andreas and Klein, 2016) and emergent communication (Lazaridou et al., 2017).

(1) The open-ended guessing game naturally elicits diverse vision-language phenomena without having to explicitly define them. Rather than instructing annotators to cover specific phenomena such as spatial reasoning or pragmatics, they emerge from the game: a describer must communicate whatever makes the target image distinguishable.

(2) The task design also rules out simple shortcuts. A model that ignores the visual input entirely cannot exceed random chance: there is only one description but ten images to choose from. Conversely, since the target image is drawn randomly from a set of highly similar candidates sharing salient features, salient visual features cannot be latched onto either. Models are forced to reason about the minimal differences between images.

3 Finally, the task proves genuinely hard. Models at the time, both zero-shot and fine-tuned directly on ImageCoDe, perform poorly, close to random chance (10%) on the video split. For example, CLIP (Radford et al., 2021a), even when finetuned on the training split of ImageCoDe, only reaches 24.3%. When we design more deeply integrated architectures that allow cross-modal and cross-frame reasoning at multiple levels, the gains are modest as well.

Our analysis reveals videos are a much richer source of minimal change and counterfactual pairs than static images: ImageCoDe's video split is significantly harder, covering more diverse phenomena and eliciting more detailed descriptions. We similarly leverage videos in the following Chapter 3 to train better image editing models. We also use the ImageCoDe benchmark in Chapter 4 to evaluate image generation models such as Stable Diffusion. Finally, in the discussions of Chapter 6, we will discuss whether models released years later, trained on orders of magnitude more data, fluent at nuanced language understanding like pragmatics, and with inherent multi-image reasoning abilities, perform better.

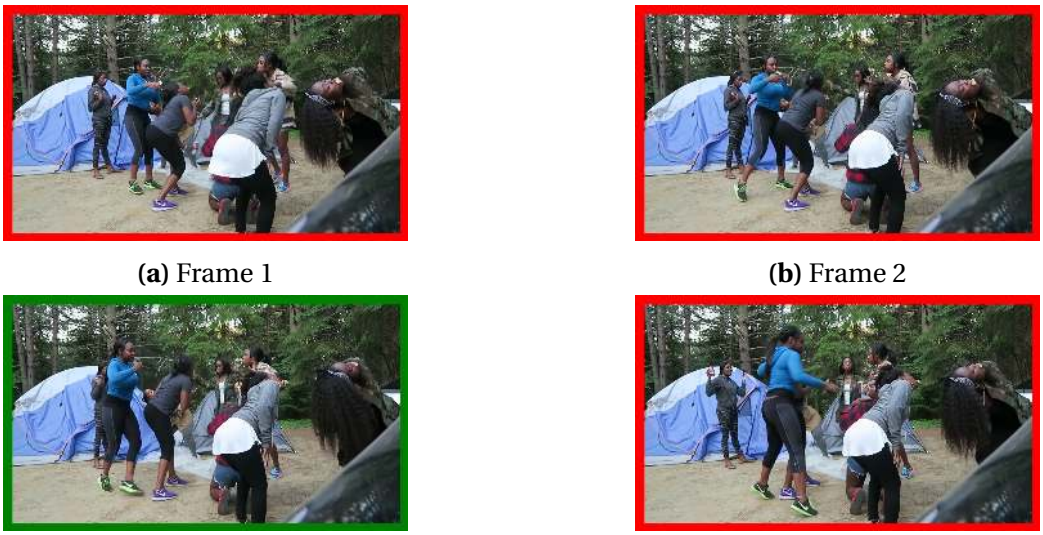
The Krojer et al. (2022) manuscript follows as originally published.

2.1 Introduction

Natural languages are highly contextual (Fodor, 2001): for a listener, recovering the speaker's *intended* meaning requires integrating information from different streams, such as grounding in perception (Pecher and Zwaan, 2005), shared world knowledge, and temporal reasoning (Wilson and Sperber, 1998). These processes, more generally, fall under the umbrella term of *pragmatics* (Grice, 1957). Despite recent progress in multimodal systems, it remains unclear to which extent they can handle settings where context plays a major role, such as in real-world communication.

To this end, we present a new challenge that requires multimodal models to leverage context to retrieve images from text. In particular, given a contextual description and a set of minimally contrastive candidate images, i.e. differing

only in some details, the model has to retrieve the target image. In order to discriminate between similar images, human annotators naturally produce highly nuanced and grammatically complex descriptions. An example of our new challenging dataset, Image Retrieval from Contextual Descriptions (IMAGECODE), is shown in Figure 2.1.



(a) Frame 1 **(b)** Frame 2

(c) Frame 3 **(d)** Frame 4

Figure 2.1: An example of the new challenge, Image Retrieval from Contextual Descriptions (IMAGECODE): *"The girl in blue is to the left of the girl in the middle with the purple shoes. The girl in blue is not obscured in any way."* Frames 5–10 are left out for simplicity's sake. The target image, frame 3, is in green, whereas the incorrect frames are in red.

During the data collection process, sets of similar images are selected among static pictures from Open Images (Kuznetsova et al., 2020) and (a larger portion) among video frames from diverse domains. Including both types of images allows for diversifying the dataset while representing different degrees of visual similarity within each set. Next, we crowdsource a *contextual* description of a target image (presented together with the rest of the set) that contains only differences relevant for retrieval. After a filtering phase involving human retrievers, we obtain a large-scale dataset with 94,020 images and 21,202 descriptions

associated with image sets of size 10.

As a result of this annotation protocol, successfully completing the task requires models to integrate several kinds of context: **i**) the image set, as the descriptions often only make sense in the context of several other images and are not suitable as stand-alone captions. In fact, aspects of the image that are very salient and that therefore would normally be emphasized are not useful in our proposed task. Instead, the focus of our descriptions are fine-grained details that help discriminate between images (see Figure 2.1); **ii**) the speaker's intention. Due to their high degree of image similarity, contextual descriptions may be literally true for multiple images; however, once the speaker's intention is taken into account, the correct image can be determined by virtue of pragmatics, i.e. Grice's maxim of quality [2] (see Figure 2.2, Figure B.3); **iii**) temporal sequences: for video frames temporal reasoning is also required to compare different moments of an unfolding event.

On our new dataset IMAGECODE, we benchmark a series of vision-and-language models that achieve state-of-the-art performance on other multimodal tasks, specifically ViLBERT (Lu et al., 2019) and UNITER (Chen et al., 2020b) as two cross-encoder variants and CLIP as a strong bi-encoder (Radford et al., 2021b). We report several findings. First, accuracy on static images is vastly superior than on video frames. Therefore, the degree of similarity among the candidate images has an overwhelming impact on retrieval performance. Second, all state-of-the-art models generally struggle with image retrieval from contextual descriptions, whereas humans consistently achieve high accuracy.

Hence, we propose model variants capable of better taking context into account: **i**) once an image-description pair is encoded, we refine this representation by attending to the other images in the set; **ii**) we augment image encodings with temporal embeddings. Based on our results, models take advantage of this additional information fruitfully but only to a limited degree.

[2] Note: While we do not model pragmatics explicitly in our baselines, we find that the IMAGECODE contains many examples suitable for pragmatic modeling

Because of its challenging nature, due to the minimally contrastive images and complex descriptions, we believe that IMAGECODE will help make visio-linguistic models more context-aware and sensitive to fine-grained details.

2.2 Related Work

There is a long tradition of grounding language understanding on single images, in the form of visual question answering (Goyal et al., 2017b; Hudson and Manning, 2019), visual dialogue (de Vries et al., 2017; Das et al., 2017), or visual entailment (Xie et al., 2019). Recently, more and more focus has been directed to settings where the visual context consists of multiple images, either conventional static pictures (Vedantam et al., 2017; Hu et al., 2019; Suhr et al., 2019; Forbes et al., 2019; Hendricks and Nematzadeh, 2021a; Yan et al., 2021; Hosseinzadeh and Wang, 2021; Bogin et al., 2021; Liu et al., 2021), or video frames (Jhamtani and Berg-Kirkpatrick, 2018a; Bansal et al., 2020). While many of these benchmarks involve just two images, COVR (Bogin et al., 2021) and ISVQA (Bansal et al., 2020) provide more images, similar to our sets of 10 images.

ISVQA and Spot-the-diff (Jhamtani and Berg-Kirkpatrick, 2018a) are most similar to our dataset, IMAGECODE. ISVQA is based on several video frames that are synthetic and cover a restricted domain, with short questions for Visual Question Answering. Spot-the-diff provides two frames from surveillance video cameras and descriptions of all their differences. IMAGECODE is unique as a) we cover a wider range of domains; b) we construct image sets that are maximally similar while being distinguishable through natural language (Section 2.3) and c) we limit descriptions to *relevant* differences. This results in (a) diverse, (b) complex and (c) pragmatically informative descriptions.

We do not claim to explicitly model pragmatics in this paper, i.e. with *Rational Speech Acts* (Goodman and Frank, 2016). Instead we present a dataset that is naturally suitable for pragmatic reasoning (Andreas and Klein, 2016; Cohn-Gordon et al., 2018) as a listener has to consider the context, assume a Gricean speaker and resolve ambiguities resulting from nuanced differences. The rea-

soning in our task and data collection is therefore also similar to ReferItGame and subsequent work (Kazemzadeh et al., 2014; Mao et al., 2016) where one crowdworker generates a referring expressing for an object in a single image and another worker picks an object based on the expression.

2.3 Data Collection

Our data collection involves two steps with a human describer and retriever. The describer is given a set of 10 highly similar images $S = [I_1, I_2, ..., I_{10}]$, one of them marked as the target image I_t, and has to write a description D that clearly distinguishes I_t from the other distractor images. In the second step, the retriever is given the same 10 images and the description from the first step and has to identify the target image based on the description. S and D are only added to our dataset if the retrieval is successful.

Below, we outline the main stages of data collection: first, the collection of similar, contrastive images in Section 2.3.1. Then, the crowdsourcing of contextual descriptions in Section 2.3.2 and validation of the examples via image retrieval (Section 2.3.3). The final IMAGECODE dataset consists of 94,020 images (partitioned into 9,402 sets) and 21,202 contextual descriptions (16,594 in the train split, 2,302 and 2,306 in the validation and test split respectively).

2.3.1 Collecting Similar Images

In the first stage, we collect sets of images that are highly similar but still distinguishable from each other by a human. To quantitatively measure the pairwise similarity of two images, we compute the Euclidean distance between their encodings extracted from a pre-trained CLIP model (Radford et al., 2021b).[3] To study the effect of different degrees of similarity, further variegate our dataset, and enable temporal reasoning, we source our candidate images from collections of static pictures as well as videos, as detailed below.

[3]We also experimented with ResNet-50 features, but we found CLIP results to be more similar to that of humans in preliminary experiments.

Static Pictures. We obtain image sets from one of the largest repositories of static pictures, the Open Images Dataset V6 (Kuznetsova et al., 2020), containing 1.74M images. For each image, we retrieve the 9 closest images from the training set based on their CLIP encodings. We then randomly sample 4,845 of these image sets.

Video Frames. As sources for our video frames, we use **i**) Video-Storytelling (Li et al., 2019), covering social events (wedding, birthday, Christmas, camping); **ii**) general-domain MSR-VTT (Xu et al., 2016); and **iii**) YouCook (Das et al., 2013), covering cooking events. We choose these datasets as they contain publicly available and general-purpose videos (not specific to downstream tasks). We retain the original splits for train, validation, and test.

To obtain disjoint sets of 10 similar frames, we first segment the videos into smaller scenes (also known as shots) via the scene detection functionality of `ffmpeg` (Tomar, 2006). Then, for each scene, we add its first frame to the set of selected images. We then iterate over every following frame and add it to the set if its pairwise Euclidean distance with each of the previously selected frames is larger than a threshold.[4] Once the set contains 10 images, we reiterate the procedure for a new set. If the scene ends and the current set contains less than 10 images, the set is discarded.

During this process, we additionally remove frames that **i**) are too blurry, i.e. their BRISQUE score (Mittal et al., 2012) is larger than 0.65; or **ii**) contain too much text, which is detected with the OCR tool Tesseract (Smith, 2007).[5] We use all of YouCook's image sets and (due to cost constraints) randomly sample image sets from Video-Storytelling and MSR-VTT for crowdsourcing (cf. Table 2.1). We remark that image sets are further filtered at the final stage of annotation (Section 2.3.3).

[4]The distance threshold was manually chosen as 0.35 based on qualitative results.

[5]The rationale of the second criterion is to prevent workers from focusing on the overlaid text rather than image content.

Dataset	After §2.3.1	After §2.3.3
MSR-VTT	11,643	8,045
Video-Storytelling	11,459	8,153
YouCook	894	588
Open Images	4,845	4,416

Table 2.1: Number of descriptions from each source of images at different stages of the annotation process.

2.3.2 Crowdsourcing Contextual Descriptions

After creating sets of highly-similar images in Section 2.3.1, we request annotators from Amazon Mechanical Turk (AMT) to write contextual descriptions for each target image in a set. Each round, a set of images is presented in random order for static pictures and respecting temporal order for video frames. This encourages annotators to take the dynamics of the event into account. We then (randomly) select 3 target images per set, and ask annotators to produce a description that discriminates them from the other images in the set. To encourage pragmatic reasoning, we do not ask for *all* the differences (just those sufficient for retrieval) and do not allow explicit mentions of other images (see Figure 2.2). We select high-quality annotators according to criteria in Section B.2 and assign partly disjoint sets of annotators to train and test in order to avoid annotator bias (Geva et al., 2019).[6]

2.3.3 Human Validation via Image Retrieval

Finally, we validate the annotation crowdsourced in Section 2.3.2 by asking AMT workers to retrieve the correct target image from a set given its contextual description. For the final dataset, we retained only the examples that i) were retrieved successfully in the training set by a single worker or ii) were retrieved successfully by *at least* 2 out of 3 workers in the validation and test sets. As a

[6]For further details on crowdsourcing instructions, analysis of annotator bias and the AMT interface, please refer to Section B.3 and Section B.4.

Phenomenon	all %	videos %	static %	Example from IMAGECODE	Definition
Context	47.3	**57.3**	6.6	Figure 2.2	Visual context or pragmatic inference required.
Temporal	15.0	**18.5**	4.1	*A smiling boy just **begins to** look towards the dog.*	Temporal markers (e.g., *after*) and verbs (e.g., *starts*)
Quantities	48.5	47.7	**51.0**	*There is an **equal amount** of yellow and white between **both** hands.*	—
Spatial Relations	70.5	**72.2**	65.3	*The cloud on **top left side** of box only has **half of** it showing.*	—
Negation	17.9	**20.7**	6.1	*The spoon is at the top right corner, it is **not** moving any of the food.*	—
Visibility / Occlusion	45.5	**54.5**	8.6	*The flowers the woman in the teal strapless dress is carrying are **completely obscured** by the man in the black shirt's head.*	An entity is covered or partially outside of the image.
Nuances	26.3	**31.6**	5.1	*There is the **slightest of openings** to see the end of the bridge through the obstruction.*	Description grounded on small patch of pixels or very non-salient aspects.
Co-reference	41.5	**42.4**	38.8	*The cloud on top left side of box only has half of **it** showing.*	—
Meta Properties	12.0	**13.9**	6.1	***Bright shot** of a girl and boy standing up straight. Her eyes are closed.*	Blurriness, brightness, overlays, and transitions of frames.

Table 2.2: Distribution of challenging phenomena in IMAGECODE based on 200 (or 1000 if underlined) manually annotated examples.

consequence, we filtered out 26.5% of the contextual descriptions generated in Section 2.3.2. Table 2.1 compares the number of examples retained at each stage throughout the dataset creation.[7]

(a) Frame 1 & Frame 2 **(b)** Frame 3 **(c)** Frame 4 & Frame 5

Figure 2.2: An example with description: "*No bridesmaid visible at all.*". Visual context is necessary to identify the correct target image, by cross-referencing the portions of images with bridesmaids (red boxes).

2.4 Data Analysis

2.4.1 Human Accuracy and Agreement

To quantify the reliability of the process outlined in Section 2.3, we report the inter-annotator agreement on our final dataset in Table 2.3. We use Krippendorff's α as a metric (the higher the better), which accounts for incomplete data, since the number of annotators per example is not fixed. We treat the index of the target image either as a nominal variable for static images or as an ordinal variable for video frames. In both cases, we find a high degree of agreement. Moreover, in Table 2.3, we also report human accuracy– the percentage of times an annotator retrieved the correct target image from a contextual description (as described in Section 2.3.3). This provides an upper ceiling for the model performances (see Section 2.6).

[7]Again, the set of workers validating train and test sets were partly disjoint to avoid annotator bias.

Metric	val	test
Human Accuracy	90.9	90.8
Krippendorff's α (nominal)	.797	.795
Krippendorff's α (interval)	.872	.869

Table 2.3: Human performance and inter-annotator agreement on IMAGECODE.

	ours	NLVR2	Spot-the-diff
Average length	23.3	15.3	10.6
Word types	6,916	6,602	2,282
Average tree depth	5.1	4.8	4.3
Average sentences	1.6	1.0	1.0

Table 2.4: Text statistics of IMAGECODE vs. other vision-and-language datasets.

2.4.2 Language Statistics

In Table 2.4, we measure a series of statistics of the descriptions collected for IMAGECODE and compare them with other vision-and-language datasets with multiple naturalistic images (cf. Section 2.2), such as NLVR2 (Suhr et al., 2019) and Spot-the-diff (Jhamtani and Berg-Kirkpatrick, 2018b).[8] In particular, we count the average description length, the number of distinct word types, the average dependency tree depth of each sentence,[9] and the average number of sentences per description. Based on these metrics, we find evidence that IMAGECODE's descriptions are longer and more syntactically complex than in the other datasets. Moreover, they include multiple sentences (11.8% of examples have 3 or more).

2.4.3 Vision Statistics

By calculating the average pairwise Euclidean distance between CLIP-based encodings of images in the same set, we find that video frames are more similar than static pictures – as expected – by a factor of 1.13. Moreover, we find that descriptions of video frames mention human body parts (72.1%) more often than static pictures (30.2%). On the other hand, names of colors appear in descriptions of static pictures (61.4%) more frequently than video frames (33.6%).[10]

[8] For comparability, we measured the statistics for all the datasets with the same tools.

[9] We use spaCy (Honnibal and Montani, 2017) as a parser.

[10] We calculated these percentages based on a list of 171 body parts in English collected by Tjuka (2021) and a list of colors in English from games4esl.com.

Thus, annotators resort to different strategies to discriminate between different types of image sets, focusing on the aspects that vary the most.

2.4.4 Challenging Phenomena

Finally, we identify 9 interesting and challenging phenomena in IMAGECODE and annotate whether they are present in 200 examples from the validation set. We provide the definition of each phenomenon, its frequency, and an illustrative example in Table 2.2. An example for each phenomena is given in Section B.7. For 4 of these phenomena unique to IMAGECODE, we further annotated 800 examples for the purpose of error analysis in Section 2.6. Inspecting these examples, we find a high number of cases where the visual context (47.0%) is required to complete the task. For instance, consider Figure 2.2: the description *"No bridesmaid visible at all."* requires a retriever to resolve the co-references of the entities in 5 frames. In particular, the body parts of the bridesmaids (red boxes) visible in frames 2 and 4 would not be identifiable as such without frame 1 and 5, respectively (where they appear with matching dresses and flowers in their hands). A common example we find in the data are "gradable" scenarios, i.e. "*The person is looking down*" might be semantically true for more than one image but it fits best to the image where the person is looking down the most.

Another group of phenomena characteristic for IMAGECODE originates from its minimally contrastive setup: annotators might focus on how an event unfolds over time (*temporal context*), on what is missing in a specific frame but visible in the others (*negation*), on what moved out of frame (*visibility / occlusion*), or on small regions and patches of pixels (*nuances*). Importantly, these phenomena are less prominent in static pictures than in video frames (cf. Table 2.2).

2.5 Methods

2.5.1 Baselines

In order to assess whether vision-and-language models can retrieve the correct image from a contextual description on a par with humans, we benchmark

three state-of-the-art models that represent three main families of multimodal architectures (Bugliarello et al., 2021; Miech et al., 2021): **i**) ViLBERT, a cross-encoder where language and vision streams can interact via cross-attention at intermediate layers (Lu et al., 2019); **ii** UNITER, a single-stream encoder where language and vision tokens are concatenated as inputs and processed with a single Transformer (Chen et al., 2020b); **iii**) CLIP, a bi-encoder where language and vision streams are independent (Radford et al., 2021b). It is worth noting that ViLBERT and UNITER are more expressive due to their architecture, whereas CLIP boasts a higher parameter count, is pre-trained on a larger dataset and uses a contrastive objective.

We evaluate these models under two different regimes: i) *zero-shot* inference, where pre-trained models are deployed on the IMAGECODE test set directly; and ii) *fine-tuning*, where the models are refined on the full training set before evaluation. We cast the training objective as binary classification for ViLBERT and as 10-class classification for CLIP.[11] Crucially, in both cases, positive and negative examples during training are sampled at random independently from the image set they belong to (see the first column of Figure 2.3). Thus, the visual context of the other images in a set is only indirectly accessible at inference time, where the image with the highest probability is predicted.

2.5.2 Integrating Context into Vision-and-Language Models

For the fine-tuning regime, we further investigate some modifications in the training setup and model architecture that facilitate the integration of visual and temporal context into the model. First, we use an alternative objective where all three models are trained on 10-class classification, but the 1 positive and 9 negatives are sourced from the same image set. The consequence of including positive and negative examples from the same image set in the same mini-batch is providing a wider visual context. We refer to this variant as +CONTEXTBATCH (second column of Figure 2.3).

[11] We found this solution to work better for each model in practice, which is justified by their different pre-training objectives.

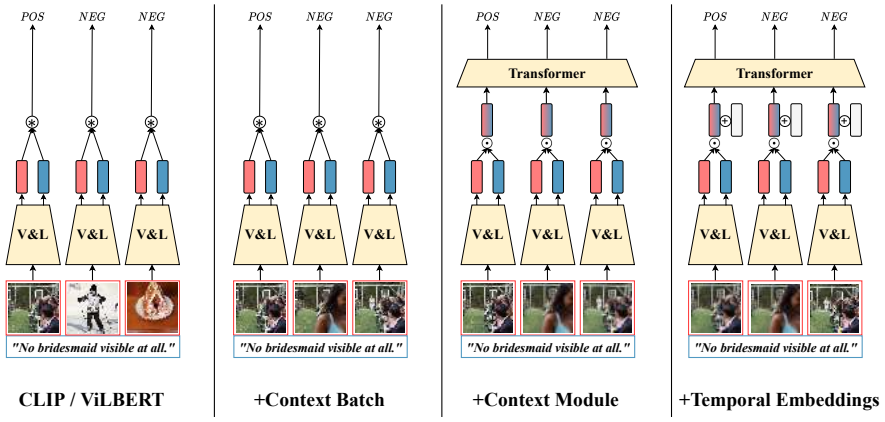


Figure 2.3: Models with increasing levels of context integration: see Section 2.5 for more details. In the figure, we colour visual embeddings in red, text embeddings in blue, and positional embeddings in grey. *POS* is the score for the target image and *NEG* for the other candidates. $\circledast$ represents dot product for CLIP and element-wise multiplication followed by a linear layer for ViLBERT/UNITER. $\odot$ represents element-wise multiplication. For ease of exposition, we show 3 images instead of 10.

This setup only conveys the visual context as a weak signal, since the model has no chance to directly compare the images in the same set. Hence, we experiment with enhancing the architecture of vision-and-language models with a mechanism inspired by Bogin et al. (2021). In particular, given an encoder (CLIP, ViLBERT or UNITER), we obtain the representations of a contextual description $\mathbf{x}_L \in \mathbb{R}^e$ (where e is the model hidden size) and of the images in a set $(\mathbf{x}_V^{(1)}, \ldots, \mathbf{x}_V^{(10)}), \mathbf{x}_V^{(i)} \in \mathbb{R}^e$ from their final layer.[12] Then, we create a series of multimodal embeddings via element-wise multiplication: $(\mathbf{x}_L \odot \mathbf{x}_V^{(1)}, \ldots, \mathbf{x}_L \odot \mathbf{x}_V^{(10)})$. Finally, we feed these to a l-layer Transformer $\mathrm{Tf}: \mathbb{R}^{10\times e} \rightarrow \mathbb{R}^{10\times e}$ to obtain context-aware multimodal embeddings $(\mathrm{Tf}_{(1}, \ldots, \mathrm{Tf}_{(10)}$. Since each description–image pair can now attend on the others in a set, the model can fully exploit the visual context. We obtain the score for the i-th pair through a linear classifier head

[12] We use the CLS tokens for UNITER/ViLBERT.

$W \in \mathbb{R}^{1\times e}$. The target image is predicted as

$$\arg\max_i \text{softmax}\left[W\left(\text{Tf}(_i+(i))\right)\right] \tag{2.1}$$

Note that we add a highway layer from the input to the output of the Transformer. We label this model variant +CONTEXTMODULE.

Finally, in addition to visual context, we make models aware of the temporal context too, as shown in the fourth column of Figure 2.3. For video-based examples only, the multimodal embeddings of each description-image pair are summed with a learnable positional embedding $\mathbf{t} \in \mathbb{R}^e$ that reflects the temporal order of the frames.[13] Thus, $(\mathbf{x}_L \odot \mathbf{x}_V^{(1)} \oplus \mathbf{t}^{(1)}, \ldots, \mathbf{x}_L \odot \mathbf{x}_V^{(10)} \oplus \mathbf{t}^{(10)})$. Multimodal embeddings are then fed to a Transformer as above. We label this variant encapsulating both visual and temporal context +TEMPORALEMBEDDINGS.

2.5.3 Experimental Setup

For all CLIP experiments, we use a pre-trained model with the vision backbone VIT-B/16[14]. We train the full models with a batch size of 360 examples (i.e., 36 image sets) for CLIP and 150 examples for ViLBERT/UNITER. We perform early stopping based on the validation accuracy with a maximum of 30 epochs. In the variants that adopt the base version of a model, we select a learning rate of $4\cdot10^{-6}$ for CLIP, $5\cdot10^{-6}$ for ViLBERT, $4\cdot10^{-5}$ for ViLBERT+CONTEXTBATCH, $8\cdot10^{-6}$ for UNITER, and $7\cdot10^{-6}$ for UNITER++CONTEXTBATCH. We find these values via hyper-parameter search on the range $[10^{-4}, 10^{-7}]$.

For CLIP variants that modify the model architecture, we adopt the following setup: first, we fine-tune the full model in the +CONTEXTBATCH regime as detailed above. Afterwards, we freeze the encoder parameters and train the components responsible for processing the multimodal embeddings, described in Equation (2.1). More details are provided in Section B.6. For ViLBERT and UNITER we finetune the whole architecture at the same time.

[13] In the examples with static pictures, no temporal embedding is added.

[14] `https://github.com/openai/CLIP`

All descriptions in IMAGECODE exceeding the maximum length of the three models are truncated. Due to their negligible amount, this does not affect performance significantly.

2.6 Results

In Table 2.5, we report the performance of the models from Section 2.5 for all the test examples in IMAGECODE as well as for the subsets containing only video frames or static pictures (see Section B.5 for validation scores). Note that the random chance baseline has an accuracy of 10%. In what follows, we compare the results across several dimensions.

Zero-shot vs. fine-tuning. In the zero-shot setting, we observe that CLIP representations are surprisingly superior to UNITER/ViLBERT even though CLIP has separate streams to encode an image and its description. In the simplest fine-tuning setting (i.e., if negatives are randomly sampled independent of the image set), we find that overall there is only a small increase in performance compared to zero-shot inference. This demonstrates that in the regime where images in the same set do not appear in the same batch during training, models cannot extrapolate how to leverage the visual context at inference time.

Adding context. For the fine-tuning regime, we observe instead a different trend once the visual context of the other images in a set is provided during training (+CONTEXTBATCH): CLIP and UNITER receive a significant boost in performance (i.e. +14.4% for CLIP), which is particularly accentuated for static pictures. On the other hand, ViLBERT's performance remains the same. Stacking a special module for contextualizing multimodal representations on top of the encoders (+CONTEXTMODULE), instead, yields gains for ViLBERT compared to +CONTEXTBATCH, whereas CLIP and UNITER are unaffected (slight drop). This shows that all models can exploit visual context, but different strategies (contrastive training or dedicated modules) may be necessary.

Finally, all three models achieve the highest performance when fine-tuned with both visual and temporal context. Adding temporal positional embed-

	all	video	static
	ZERO-SHOT		
CLIP	22.4	15.6	47.8
	FINE-TUNING		
CLIP	24.3	17.1	51.3
+CONTEXTBATCH	28.4	20.0	**60.0**
+CONTEXTMODULE	27.7	19.6	58.4
+TEMPORALEMBEDDINGS	**29.9**	**22.0**	59.8
	ZERO-SHOT		
UNITER	19.8	13.6	42.9
	FINE-TUNING		
UNITER	21.9	14.4	50.1
+CONTEXTBATCH	24.8	17.4	52.8
+CONTEXTMODULE	24.4	16.7	**53.0**
+TEMPORALEMBEDDINGS	**25.7**	**19.1**	50.5
	ZERO-SHOT		
ViLBERT	19.3	13.5	40.8
	FINE-TUNING		
ViLBERT	20.9	13.1	**49.9**
+CONTEXTBATCH	20.9	15.0	42.7
+CONTEXTMODULE	22.3	16.1	45.6
+TEMPORALEMBEDDINGS	**24.5**	**18.0**	49.3

Table 2.5: Performance (test accuracy) on IMAGECODE across two training regimes (zero-shot and fine-tuning), three models (CLIP, UNITER, ViLBERT) and 4 model variants. We report separate figures for all the examples and two disjoint subsets: video frames and static pictures.

dings on top of the contextual module (+TEMPORALEMBEDDINGS) yields an accuracy of 29.9 for CLIP, 25.7 for UNITER and 24.5 for ViLBERT. Crucially, even the best-performing models lag significantly behind the (micro-averaged) human accuracy of 90.8 (cf. Table 2.3). Hence, despite some limited ability to integrate context, models are currently incapable of the fine-grained reasoning and pragmatic inferences needed to solve IMAGECODE.

Pre-trained model. Across all model variants and training regimes, CLIP consistently achieves higher accuracy than ViLBERT or UNITER. This implies that a larger amount of parameters, pre-training examples or the contrastive objective are more beneficial than ViLBERT's or UNITER's more expressive model architecture. Thus, these results violate the expectations that attention between vision and language would be more suitable to jointly encode highly nuanced visual details and descriptions (Miech et al., 2021). Additionally UNITER slightly outperforms ViLBERT as its single-stream architecture might enable richer cross-modal interactions.

Video frames vs. static pictures. The highest accuracy on the subset of the data with video frames (20.9) is far lower than that for static pictures (59.4). This confirms that videos represent the main challenge in IMAGECODE, both because of the higher similarity of images in a set and of the particular factors of variation that help differentiate among them (cf. Section 2.4.3 and examples in Section B.7). Additionally, model performance on video frames seems to increase more consistently as more context (both visual and temporal) is provided, whereas there is no clear trend in the case of static pictures.

Error Analysis. On a broad level, we have seen that video frames are much more challenging for models. Next, to identify more fine-grained causes for the overall low performance of the vision-and-language models on IMAGECODE, we compute the Pearson's correlation between accuracy and a series of possible explanatory variables. In particular, we find a weak negative correlation with the number of tokens in the description ($r = -0.11$) and a weak positive correlation with the average pair-wise Euclidean distance between CLIP encodings of the images in a set ($r = 0.22$), which represents visual similarity.

By focusing on the 1000 annotated examples in Table 2.2 we observe a stark drop from overall performance on the subset of examples containing nuances, visibility/occlusion, and negation (Figure 2.4). This confirms insights from Kassner and Schütze (2020) and Hosseini et al. (2021) on the difficulty of modeling negation in text-only models.

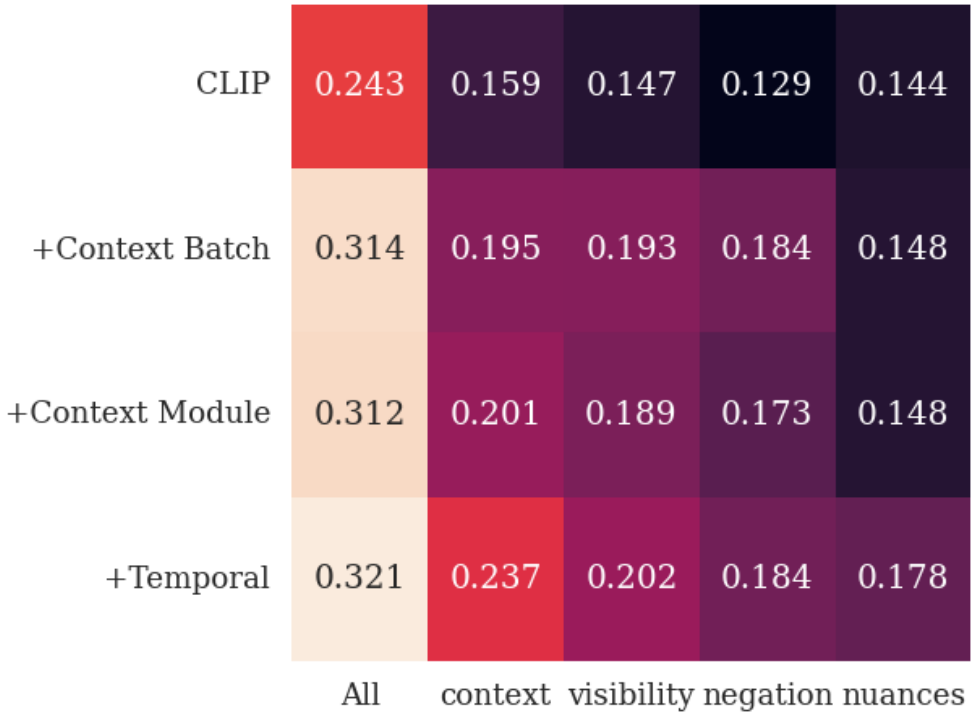


Figure 2.4: Performance of different CLIP variants (rows) on subsets of examples containing phenomena of interest (columns) in 1000 annotated validation examples. The hue of each cell indicates accuracy.

2.7 Conclusions and Future Work

We created a new challenge, Image Retrieval from Contextual Descriptions (IMAGECODE), which is designed to evaluate the ability of vision-and-language models to integrate visual, pragmatic, and temporal context into their predictions. In particular, given a complex and nuanced *contextual* description, a model is required to retrieve the corresponding image from a set of highly similar candidates. We benchmarked state-of-the-art bi-encoder and cross-encoder models, such as CLIP and ViLBERT. Moreover, we proposed new variants of these models that are more suitable to solve this task, by augmenting them with

a module to attend on the other images in a set and temporal embeddings. We found that IMAGECODE is highly challenging for all variants: even the best model (28.9) lags behind human performance (90.8) dramatically. Images sourced from video frames display the largest gap in performance. The most challenging phenomena in IMAGECODE include pragmatics, negation, fine-grained distinctions between images, and occlusion among others.

2.8 Acknowledgements

IMAGECODE wouldn't have been possible without the herculean effort of the Amazon Mechanical Turkers and their feedback on the interface. We also thank Emanuelle Bugliarello for his help with VOLTA, an excellent codebase for several vision and language models. We thank the members of SR's research group for their feedback on the ideas presented here. IMAGECODE is funded by the Mila-Samsung grant program. We thank Microsoft for providing us Azure credits. SR acknowledges the support of the NSERC Discovery Grant program and the Facebook CIFAR AI Chair program.

2.9 Ethics and Limitations

We distribute the descriptions in IMAGECODE under MIT and adopt the licenses of the video and image sources on which our image sets build on top. We report details about crowdsourcing such as payment and selection criteria in Section 2.3.2 and Section B.2. For the tested model variants, we only train a single run for each hyperparameter setting due to long run times.

3

Learning Action and Reasoning-Centric Image Editing from Videos and Simulations

Preface

In the previous Chapter 2 I have shown what videos can bring to the table, beyond static images. And I have shown how, in general, the paradigm of rigorously curating minimal pairs can be fruitful and lead to high-quality data, isolating the aspect that we want models to learn or be evaluated on. This chapter also marks the thesis's shift from the behavioral evaluation of Chapter 2 to its *modeling* angle on deep integration. In this chapter, published at NeurIPS 2024 (Krojer et al., 2024b), we continue with the two themes (video-sourced data and the minimal pairs paradigm) on the task of text-guided image editing (Brooks et al., 2023): a

task that bridges two traditionally separate paradigms in computer vision (visual understanding and visual generation), requiring a single model to master both. We observe that existing image editing models are trained on noisy data, i.e., the source and target images exhibiting more changes than the prompt describes. We also observe that existing models perform poorly on action and temporal edits, and the reason is largely a data problem: object and attribute edits can be learned from static images, but action and reasoning-centric edits require data from entirely different sources covering physical dynamics, temporality and spatial reasoning.

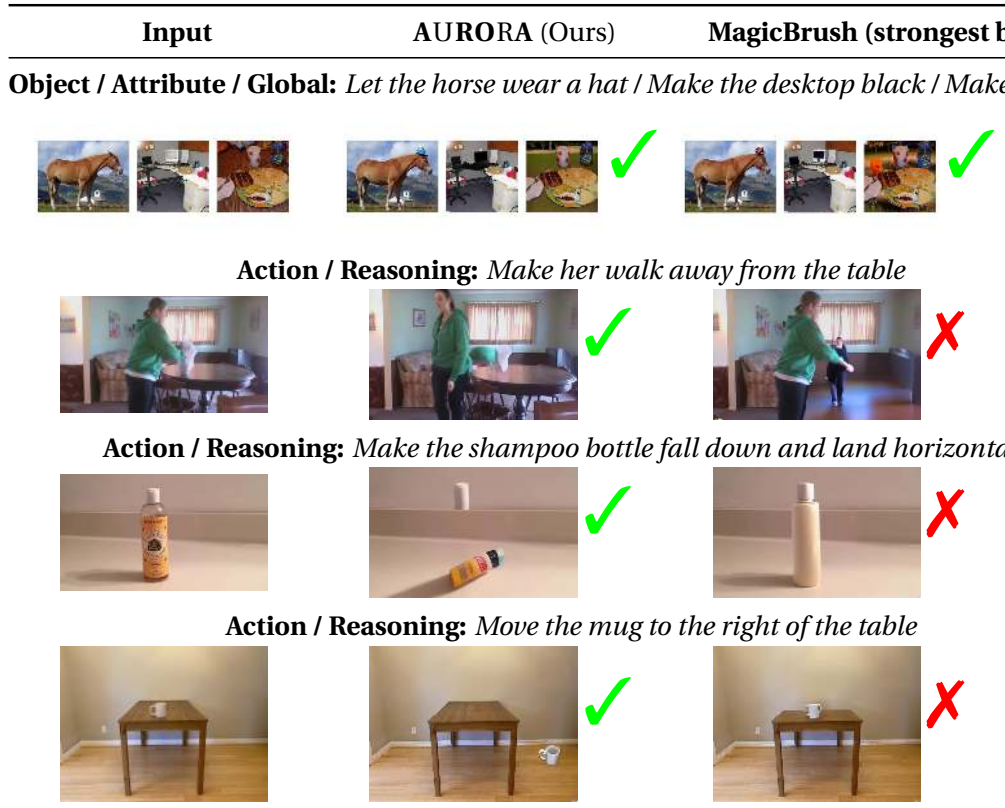
To achieve a better and more general-purpose image editing model, we propose AURORA in this chapter, a minimal change image editing training dataset that covers all major types of image edits. Crucially, following the minimal pairs philosophy of ImageCoDe, each training example contains exactly one meaningful visual change, making the supervision signal as precise as possible. By training on the 289K AURORA examples, we set a new state-of-the-art[1] on a curated benchmark ranging from simpler object edits to complex reasoning and action edits. Finally, we also critically examine evaluation practices at the time, revealing that trivial "copy the source image" baselines can hack CLIP-based metrics.

This chapter studies a task that *explicitly* requires models to unify visual understanding and visual generation, while also of course reasoning about the language prompt. In the next chapter (Chapter 4) we will study if a model solely trained on visual generation can *implicitly* also perform visual understanding tasks.

The Krojer et al. (2024) manuscript follows as originally published.

[1] at the time of publication in May 2024.

Figure 3.1: Previous failures on editing skills such as action, movement and reasoning (measured in **AURORA**-BENCH) compared to improvements with **AURORA** on these more challenging actions.



3.1 Introduction

Image editing is a complex task, involving many different skills, from adding/removing objects, changing colors/textures/styles, to **"taking actions": moving objects, changing actor positions or even more complex interactions**. Tackling all of these requires fine-grained understanding of how visual scenes are composed as well as **reasoning** (e.g. spatial instructions or referring expressions). No current model can successfully do all of these edits, and most only perform localized changes involving object addition/removal or attribute edits, following the "inpainting paradigm" (Zhang et al., 2024; Xie et al., 2023). Others have

tried to address this issue by introducing more specialized model architectures which handle different editing subtasks (Couairon et al., 2023; Zhang et al., 2023a). However, **neither of these approaches includes edits requiring more holistic visual understanding of how humans and objects interact or how events unfold**, such as 'make the cook cut the apple in half' or 'make the dog jump in the air' (see Figure 3.1). These more *action-centric* edits are severely understudied in the space of instruction-tuned image editing models (Brooks et al., 2023; Huang et al., 2024); when they are considered, it is done in isolation, ignoring other image edit subtasks and rigorous semantic evaluation (Souček et al., 2024; Black et al., 2024). In Section 3.2 we describe a typology of these edit types and how existing datasets currently fail to address them all.

As we argue in this paper, **a major reason for these limitations is the lack of high-quality data**. Finetuning data of object or attribute changes is simpler to acquire than other forms of edits, since inpainting setups directly leverage strong object and attribute abilities of txt2img models (Rombach et al., 2022) for paired-image data generation (Yildirim et al., 2023; Zhang et al., 2024). However, solving the data scarcity for learning action and reasoning-centric edits is not as straightforward. We identify videos and simulation engines as the two most promising sources of data for these edit types. As we discuss in this paper, we find that previous models trained on "noisy" synthetic image pairs or video frames lead to poor editing abilities. Here, noisy refers to image pairs with changes not mentioned in the prompt, i.e. due to shortcomings of the automatic generation process or inherent properties of videos such as viewpoint changes and non-meaningful movement. Therefore, our main requirement of high-quality action and reasoning-centric edit examples is that they be *truly minimal*: Edited images which contain one or maximally two semantic changes described by the prompt, while all other aspects are kept exactly the same. From a diverse set of video sources and simulation engines, we curate the **A**U**RO**R**A** Dataset (**A**ction-**R**easoning-**O**bject-**A**ttribute). Via crowd-sourcing and curation we collect 130K truly-minimal examples from videos and 150K from simula-

tion engines for instruction-tuned image editing. We describe our dataset and collection process in Section 3.3.

The few image-text-alignment metrics commonly used in image editing are based on visual similarity to a groundtruth and in reality turn out to mostly measure the ability to stay maximally faithful (i.e. copying) to the source image (Zhang et al., 2024; Fu et al., 2023). Though faithfulness is an important first step to master, these metrics have almost no correlation with the model's ability to generate accurate edits, especially on action and reasoning-centric changes. Hence, in addition to the training data in **AURORA**, we introduce **AURORA**-BENCH(Section 3.4), a manually annotated benchmark covering 8 editing tasks on which we collect human judgement (Table 3.2). Inspired by work on image generation models as discriminators (Krojer et al., 2023b; Li et al., 2023a), we also describe a novel discriminative metric that assesses *understanding* and hallucination (Section 3.5.1). To demonstrate the efficacy and quality of **AURORA**, we present a state-of-the-art instruction-tuned image editing model, finetuned on **AURORA** and evaluated on **AURORA**-BENCH, which we compare to strong baselines in a set of experiments in Section 3.5.3.

In summary our contributions are: 1) The creation of **AURORA**, a new clean and varied set of image edit pairs for instruction-finetuning that encompasses more action-centric and reasoning-centric examples. 2) We present a **comprehensive benchmark** covering a variety of edit types; 3) We introduce a **novel more informative metric** beyond existing ones; 4) We provide a **state-of-the-art image editing model** based on **AURORA** with well-rounded image editing capabilities covering object-centric, action-centric, and reasoning-centric edit abilities.

3.2 Typology of image edits

There are many ways to visually change a given scene (Huang et al., 2024). **We define and focus upon five broad types of changes: *object/attribute-centric, global, action-centric, reasoning-centric,* and *viewpoint.*** Some can overlap:

Table 3.1: AURORA vs. comparable public editing datasets. See details on all data sets in Section 3.2 and Section 3.3.2, and Section 3.3 for defining *truly minimal change*. ✓ = skill is covered but to a lesser extent.

Dataset	Semantic Quality ('Truly Minimal Change')	Skill Coverage			
		Obj. / Attr.	Global	Action	Reasoning
InstructPix2Pix	**Low**	✓ / ✓	✓	X	X
HQ-Edit	**Low**	✓ / ✓	✓	X	X
GenHowTo	**Low - Medium**	X / ✓	X	✓	X
MagicBrush	**High**	✓ / ✓	✓	X	X
+ AG-Edit (Ours)	**High**	✓ / ✓	X	✓	✓
+ Something-Edit (Ours)	**Medium - High**	✓ / ✓	X	✓	✓
+ Kubric-Edit (Ours)	**High**	✓ / ✓	X	✓	✓
= AURORA (Ours)	**High**	✓ / ✓	✓	✓	✓

an action might change the attribute of an object, or reasoning can play a role in any type. Object-centric changes correspond to changes made to a specific object such as replacing it with another one, changing its attributes like color or texture, resizing it, or removing it entirely. Global edits change the overall image such as the background, style or textures. Action-centric changes correspond to changes that occur as a result of executing an action: changes in configuration of the objects, state changes of objects (e.g. cutting an apple), or pose change. Reasoning-centric changes are broadly defined as anything requiring compositionality or symbolic understanding: spatial, resolving referrring expressions ("sitting person on the far left"), negation, etc. Finally, viewpoint edits correspond to moving an egocentric camera, zooming in/out, and are the only on we do not cover in **AURORA** as they add many additional challenges: First, in videos *in the wild*, they exacerbate the already numerous changes; second, excessive camera movement can unpredictably alter the entire scene, introducing noise. Section C.3 shows examples for each type.

Coverage in existing data: We characterize four broad sources of image pairs and prompts, which influence how much certain edit types are covered in

existing training data (see Table 3.1):

1. Combining existing text-to-image models and LLMs in pipelines to automatically generate similar **synthetic** images (Brooks et al., 2023; Hui et al., 2024; Zhang et al., 2023b);
2. Providing **humans** with an image editing tool on existing images, combined with in-painting (Zhang et al., 2024), or finding existing Photoshop edits on the web (Tan et al., 2019);
3. Selecting nearby **video** frames and captioning the change via human annotators or automatically (Souček et al., 2024; Black et al., 2024; Alzayer et al., 2024);
4. Using **simulation** engines to generate pairs by precisely controlling visual changes with templated language (Michel et al., 2023; Wang et al., 2023a).

InstructPix2Pix (Brooks et al., 2023) introduced the first large-scale instruction-guided image editing dataset (313K) in a fully synthetic manner, combining GPT-3 (for prompt generation), Stable Diffusion (Rombach et al., 2022) and Prompt2Prompt (for increasing similarity of image pairs) (Hertz et al., 2022). However, this scale comes at the cost of general data quality (see random samples in Figure C.23): Often Prompt2Prompt either fails to change anything or changes far more than asked for by the prompt, and in rare cases the prompt is non-sensical. This synthetic data is sufficient, though not optimal, for global and object-centric edits. However, action-centric and reasoning-centric edits either fail in execution or are not represented. Despite using more advanced models and pipelines, we find similar issues in **HQ-Edit (Hui et al., 2024)** under close inspection (see Section C.10). **MagicBrush (Zhang et al., 2024)** addresses some of InstructPix2Pix's shortcomings, mainly the lack of truly minimal edit pairs, with a rigorous crowdsourcing protocol where humans use the inpainting feature on the DALL-E 2 (Ramesh et al., 2022) interface. This methodology produces truly minimal image pairs for object-centric edits (see Table 3.3b). The inherent limitations of inpainting become apparent with certain attribute

edits, and it is entirely unsuitable for action and reasoning-centric editing: Changing the color of a backpack via inpainting would also change its shape, size or texture. We did not find examples of action or reasoning-centric edits in MagicBrush (see Section C.10).

The landscape of actions and reasoning editing datasets is sparser: A relevant case is **GenHowTo (Souček et al., 2024)** which focuses on video frames that display actions and subsequent state changes in instructional videos. Their image pairs (and also captions) were chosen automatically, resulting in pairs that are not always minimally different due to excessive camera changes (Section C.10 for random samples). We hypothesize that though GenHowTo may initially seem better at action-centric edits, like InstructPix2Pix it will tend to over-edit and not truly comprehend instructions due to training data quality issues. Reasoning-centric (spatial/geometric reasoning, referrring expressions, negation etc.) image pairs can be most directly created via simulation engines, with the hope of Sim2Real transfer. Simulations allows precise control over location, color and even orientation (rotation, flipping) of objects. Such reasoning is rarely covered in other sources: InstructPix2Pix and MagicBrush have almost no mentions of even the simplest spatial terms such as left" or "right".

In the next section, we present **AURORA** which addresses some of the above shortcomings by using specific video sources and simulation engines to cover action and reasoning-centric edits in addition to existing quality object-centric editing data.

3.3 AURORA: A diverse and high quality image editing dataset

We present **AURORA**, a balanced dataset covering **A**ction **R**easoning, **O**bject and **A**ttribute edits, comprising a total of 289K training examples, see Figure 3.2 and Table 3.1 for details and comparison to existing datasets. Section C.10 provides 16 non-cherry picked training samples for all datasets.

Figure 3.2: Our **AURORA** dataset covers action, reasoning and object-centric edits via 4 sub-datasets.

(a) *Add a leashed dog to the hydrant*

(b) *Show his hands on the plate arranging the food*

Figure 3.3: Common failure mode of previous models trained on noisy image pairs (e.g. InstructPix2Pix and GenHowTo): Their outputs are rarely faithful to the source image due to its noisy training pairs.

3.3.1 Truly Minimal Visual Change

As surveyed in Section 3.2, many issues in previous datasets, even when they are large-scale and diverse, can be traced back to the lack of *truly minimal* image pairs. Beyond manually inspecting examples in existing dataset, the lack of faithfulness wrt. the source image and prompt can be observed in generations of InstructPix2Pix and GenHowTo: In Figure 3.3 models changed the background, color of the hydrant, etc. and in the case of GenHowTo, we tend to see "letter artifacts" from its training data (more examples in Section C.9). MagicBrush

(Zhang et al., 2024) was able to produce much better object-centric edits simply by fine-tuning InstructPix2Pix on 8.8K truly minimal image pairs. To complement it, we create a novel (and larger) set of true minimal pairs for action-centric and reasoning-centric edits. Table 3.1 compares ours to existing datasets.

3.3.2 Creating quality data for action-centric and reasoning-centric edits

We use two types of sources to construct this new dataset: video and simulation engines.

Videos cover a wide range of action-centric edits as they are an abundant source of realistic and diverse state changes (Zellers et al., 2021; Miech et al., 2019; Niu et al., 2024). However simply taking frames from any video data *in the wild* (e.g. YouTube) often leads to noisy data (see Section 3.5.4): the camera moves, too many things move at once, or the changes are simply not meaningful (i.e. easy to verbalize). Hence, we create *Action-Genome-Edit* and *Something-Something-Edit*, two new image editing datasets based on carefully selected video frame minimal pairs. Both datasets use frames from video datasets where humans were asked to do activities in or around the house through crowdsourcing which usually represents one action in isolation.

For *Action-Genome-Edit*, we select frame pairs that had a CLIP cosine distance between 0.1 and 0.4, resulting in 15K pairs (thus filtering out many pairs with camera movement). Since no automatically generated instructions could reliably describe the changes, we tightly work with crowdworkers (Section C.4) to produce accurate edit instructions, with extensive quality screening and communication. Crucially, we ensured that workers discarded examples where a) there were too many or few changes, b) the changes were hard or lenghty to verbalize, or c) the camera moved (even if slightly) or the image quality was poor. After discarding 4K examples, the final Action-Genome-Edit dataset consists of 11K examples. **For *Something-Something-Edit*** we started from the original Something Something dataset (Goyal et al., 2017a) which consists of 221K short clips

where humans perform pre-defined basic actions such as “Attaching a string to a balloon”, “Folding a cloth”, “Lifting a book with a pencil on it”, etc. Since the first and last frame of the short clips usually depict the start and end of the action, we selected them as our source and target images. We manually identify 10-15 categories of labels that don't lead to useful changes (e.g. ”Pretending to...“ where the person does not actually perform the action) and filter those out. The results is a set of 119K minimal frame pairs with high-quality simple edit instructions.

Simulation engines To perform action and reasoning-centric editing a model has to master spatial and relational reasoning, geometry, and simple movement. While videos provide some signal for learning these skills, a realistic simulation engine (Greff et al., 2022) offers full control over the arrangement and movement of objects. To teach this basic reasoning, we create *Kubric-Edit*.

Kubric-Edit contains 150K training examples which span three reasoning-centric edit skills – location changes, rotation changes, and count changes – and one object-centric edit skill – attribute changes. We build on top of (Wang et al., 2023a) who created 6K Kubric image pairs for contrastive image-text-matching, by defining more types of change and significantly extending the dataset. We manually filter and name more than 213 realistic objects from Google Scanned Objects, define templates for the edit instructions and ensure more truly minimal change. We cover actions such as *move, turn, swap, flip upside down, add/remove*, spatial configurations such as *left, right, up, down, rotation*, and attributes such as *size, shape or color*. More details are presented in Section C.5.

Thus, the **AURORA** dataset consists of four carefully selected sub-datasets coming from three sources: MagicBrush (Zhang et al., 2024) (humans equipped with an editing tool on MS-COCO), Action-Genome Edit and Something-Something-Edit (nearby video frames with collected high-quality human captions or filtered previous labels respectively), and Kubric-Edit (from the realistic simulation engine Kubric). In Section 3.5.3 we finetune InstructPix2Pix (Brooks et al., 2023) on our new dataset and thoroughly evaluate its performance across

all types of edits. Note: Before collecting new data, we naturally tried to re-use existing datasets of visual change but found them inadequate for varying reasons described in Section C.13.1.

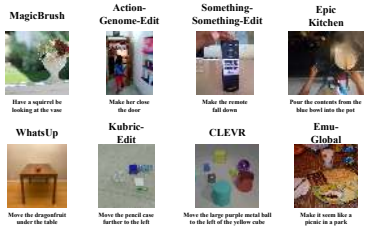
3.4 AURORA-BENCH: A holistic editing benchmark

To holistically assess the editing abilities defined in Section 3.2 (object/attribute, global, action, reasoning, excluding viewpoint), we manually **create a set of 400 image-edit-instruction pairs from 8 sources: AURORA-BENCH**. See Figure 3.4(a) for an example of each one. We ensure that **AURORA**-BENCH allows studying out-of-distribution (OOD) transfer when a model is trained on **AURORA**, e.g. Sim2Real transfer from Kubric-Edit to real-world (spatial) reasoning or action edits outside of Action-Genome-Edit or Something-Something-Edit. Each of the 8 tasks contains 50 examples of image-prompt pairs that were either directly written by the authors or manually inspected for quality.

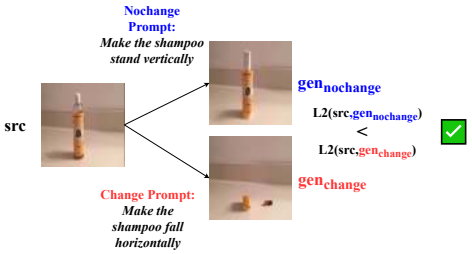
We cover object/attribute-centric edits with **MagicBrush** examples, action edits with **Action-Genome-Edit, Something-Something-Edit** and **Epic-Kitchen** (Damen et al., 2018) (OOD), reasoning-centric edits with **Kubric-Edit, CLEVR** (Park et al., 2019) (OOD) and **WhatsUp** (Kamath et al., 2023) (OOD); and **Emu-Global** covers global edits by sampling certain categories from (Sheynin et al., 2023). MagicBrush, Action-Genome-Edit, Something-Something-Edit and Kubric-Edit are introduced in the previous Section 3.3.2. We manually select Epic Kitchen frames and write prompts to study OOD generalization of action understanding since the egocentric scenes and actions are quite different from the other two action-centric subtasks. To assess transfer from our Kubric simulation data, we leverage the real-world diagnostic data in WhatsUp for spatial reasoning, and OOD CLEVR images for testing spatial reasoning in addition to complex referring expressions.

3.5 Evaluation

We begin by introducing the metrics we use on **AURORA**-BENCH. Image-editing (and thus its evaluation) can be framed as a two step process: First, given an image-prompt pair, a model must understand how they relate to each other, for example by grounding phrases in the image. This is closely related to traditional vision-and-language understanding. Second, the model must perform the required edits by generating a new image, while being faithful to the original image. Previous work evaluates this second step – the final generation, which we also adopt as our primary judgement. However much insight can be gained from assessing understanding or *discrimination* abilities of editing models present in the first step. Our second evaluation proposes a new metric that tries to measure just that.



(a) AROA-Bench: Examples from our new expert-curated benchmark covering 4 editing skills (object-centric, action-centric, reasoning, global)



(b) DiscEdit metric: The left prompt describes the source: no change needed. The right prompt is a "normal edit", requiring a change.

3.5.1 Evaluation of final generations

Existing metrics: There currently exist a series of visual similarity metrics – L1, L2, CLIP-score, DINO-score – which are commonly used to evaluate the similarity of output edit compared to groundtruth images (Zhang et al., 2024; Fu et al., 2023). The effectiveness of these metrics has not been formally justified for image editing, i.e. with human judgement correlations. We hypothesize that these metrics mostly reward models for copying information from the source image,

rather than accurate editing. This hypothesis is confirmed using a trivial baseline, simply copying the source image as its output. This *copying* model outperforms all existing models on these metrics, see Table 3.3b. For instance, using human judgements of MagicBrush and **AURORA** outputs on MagicBrush test examples, we find a very weak correlation of 0.098 (using $CLIP-I$ score), also shown in (Ku et al., 2024). Though faithfulness to the input is an important component of editing, so is actually *modifying* the image aligned with the prompt. These metrics also assume hard-to-obtain clean groundtruth images– without them meaningful automatic evaluation is even harder. Finally, since many automatic metrics rely on standard vision encoders (e.g. CLIP (Radford et al., 2021b)) trained on static images (no video data) and known to be weak reasoners (Yuksekgonul et al., 2023b), they are not suitable for our study. When we compute human correlation on WhatsUp examples (spatial reasoning), with clean groundtruth images, it drops to zero. In summary, these metrics might detect a model that struggles with faithfulness to the source image such as InstructPix2Pix (see Figure 3.3), but are not helpful for comparing the semantic accuracy of stronger models.

Human judgment of edited images: With the insight that automatic visual similarity metrics are only a weak signal, we primarily rely on human judgment of model outputs on **AURORA**-BENCH: We ask humans to rate the absolute edit success (0=none, 50=partial, 100=full) as well as comparison (i.e. win-rates) between different models. We focus on the former in our main results as it allows us to compare task difficulty. We ensure that evaluators (we pick the best three evaluators from crowd-sourcing **AURORA**) pay most attention to the correct (semantic) interpretation of the edit prompts [2]. Section C.4.2 describes guidelines, compensation and extensive communication with crowdworkers.

3.5.2 DiscEdit: discriminative evaluation of image editing

We propose an additional automatic metric *DiscEdit* applicable to **AURORA**-BENCH examples. Unlike Section 3.5.1 above which considered the overall

[2] Inspections show high-quality ratings; inter-annotator agreement is a Fleiss-Kappa score of 0.626

accuracy of generated edits, this evaluation serves as a diagnostic test for determining whether models truly understand how prompts relate to the input image, or can abstain from editing.

Inspired by text-to-image models repurposed as discriminators (Krojer et al., 2023b; Li et al., 2023a), models are given an image and two minimally different edit instructions $t_{nochange}$ and t_{change}. While $t_{nochange}$ requires *little to no change* to the source image, t_{change} requires models to perform *significant changes*. An example of such a test pair is given in Figure 3.4(b). Thus, we expect the similarity between the first generated image $i_{nochange}$ and the source image i_{src} to be higher than the similarity between the second generated image i_{change} and the source image i_{src}, which we measure as a L2 distance in the latent space of the diffusion model (written $Enc(i)$):

$$\text{Score}_{\text{DiscEdit}} = \begin{cases} 1 & \text{if } \|\text{Enc}(i_{\text{src}}) - \text{Enc}(i_{\text{pos}})\|_2 < \|\text{Enc}(i_{\text{src}}) - \text{Enc}(i_{\text{neg}})\|_2 \\ 0 & \text{otherwise} \end{cases}$$

The intuition behind *DiscEdit* is that edits should be proportional to those described in the prompt – in other words, models should not change images more than required, nor should they produce fewer edits than requested in instructions. This metric therefore tests how much models are following or 'understanding' what instructions require. On the flip side, it also quantifies a form of hallucination: Changing things even when no change is required. Since it is not possible to find a "no-change" prompt $t_{nochange}$ for all kinds of prompts, we select a subset of source-prompt pairs (i_{src}, t_{change}) from *AROA-Bench* and manually define a no-change prompt $t_{nochange}$. Section C.8 contains details on the data creation process and implementation of *DiscEdit*. A *DiscEdit* score of 0 or 1 is interpretable, and the metric does not require costly groundtruth target images. While it might seem far-fetched to expect the model to recognize when a scene does not need to be changed, we note that it is relevant in scenarios where image editing is a component of generative simulators (Yang et al., 2024b).

Table 3.2: Human Judgment of semantic editing success on **AURORA**-BENCH tasks. Humans were asked to rate the edit success from none (0), partial (50) to full (100). Extended table in Section C.7. Overall score is "balanced": we average each skill first, and then take the average of those 4 numbers. Note: Our model was trained on more of the datasets than e.g. Magicbrush, so some of them are more IID for our model.

	Obj./Attr.	Action, Human-Object-Interaction			Reasoning			Global	
Model	**Magic Brush**	**Action-Genome**	**Something Something**	**Epic Kitchen**	**WhatsUp**	**Kubric**	**CLEVR**	**Emu-Global**	**Overall Score**
GenHowTo	18.0	8.0	8.7	**17.7**	4.3	0.7	2.0	11.3	10.8
MGIE	36.0	7.0	11.3	5.0	6.0	6.7	16.0	36.5	22.5
InstructPix2Pix	31.3	13.3	12.3	4.3	0.7	5.7	14.7	33.7	20.5
MagicBrush	**61.7**	16.3	17.0	12.0	3.0	9.3	22.0	**42.3**	32.6
AURORA (Ours)	60.5	**35.6**	**31.8**	14.2	**27.3**	**59.6**	**46.1**	33.0	**41.3**

3.5.3 Results

We compare against the strongest general editing models at the time, as well as GenHowTo trained on nearby video frames (Souček et al., 2024). See Section 3.2 for details on their training data. We train our own **AURORA** model with the InstructPix2Pix architecture on the **AURORA** dataset and mix all four sources such that each dataset is equally likely to be sampled during training, and take a checkpoint that was first pretrained on InstructPix2Pix and then MagicBrush (more details: Section C.6).

Human Judgement Table 3.2 summarizes our results on **AURORA**-BENCH evaluated via human judgement of successful adherence to the prompt and source image (0=none, 50=partial, 100=full). Most notably, **our model significantly improves on the challenging action and reasoning-centric edits**. However, action edits on complex real world images remain a challenge, while we see stronger numbers on "simpler" reasoning. At the same time, **AURORA** maintains strong performance on the diverse and well-established MagicBrush test set, leading to a high *Overall Score*. Finally, we observe generalization from training on simulation to CLEVR, and notably WhatsUp, a real-world spatial reasoning task.

Table 3.3: Automatic Evaluation: DiscEdit (left) and issues with existing automatic metrics (right)

(a) Discrimination performance with *DiscEdit* (comparing the two strongest models from Table 3.2). We show binary accuracy (50% random chance). Details: we average each example over 4 noise samples for the denoising process (Section C.8)

Model	WhatsUp	Something	AG	Kubric	CLEVR
MagicBrush	0.472	0.371	0.477	0.392	0.400
AURORA	**0.565**	**0.548**	**0.583**	**0.592**	**0.450**

(b) Automatic visual similarity metrics on MagicBrush test: Naively copying (!) the input is ranked better than the MagicBrush model for which this is IID (see Section 3.5.1).

Model	L1↓ / L2↓	DINO↑	CLIP-I↑
InstructPix	0.112 / 0.037	0.746	0.8538
MagicBrush	0.072 / 0.025	0.865	0.915
Naive Copy	**0.036** / **0.015**	**0.917**	**0.943**

DiscEdit Table 3.3a shows results with the *DiscEdit* metric on **AURORA**-BENCH examples. Discriminating between minmal prompts that either require a change or no change, proves to be a hard task: With **AURORA**, performance is slightly above random on most tasks except CLEVR. Performance of the MagicBrush model is even below random chance. We investigate this surprising result but could not find any pattern. Thus, we can only hypothesize that the wording of "no-change" prompts might be more unusual, and hence lead to hallucination behaviour (example outputs in Section C.9.2). We also find several encouraging qualitative results from our model (Figure 3.4(b)).

3.5.4 Ablations and qualitative analysis

Can we quantify Sim2Real transfer? **AURORA** contains many simulated examples featuring spatial reasoning. To quantify the transfer to *spatial reasoning on real images*, we train a model on **AURORA** minus Kubric and manually rate the outputs on the WhatsUp examples from **AURORA**-BENCH: A win-rate of 46% for the full **AURORA** model, while without Kubric only a single win (2%) is found [3].

[3]The rest are ties where both fully fail.

We also find that training on truly minimal Kubric examples ”stabilizes“ training: Without it, the model hallucinates more un-needed changes and artifacts (e.g. adding people).

EditDAAM: We adopt DAAM (Diffusion Attentive Attribution Maps) (Tang et al., 2023) for qualitatively studying the attention maps of our editing model but study patterns across U-Net layers, grouping them into *Down*, *Middle* and *Upper* layers. We intuit that image *understanding* happens in earlier layers and the final *generation* in later layers. Since our model has seen more movement-based and spatial edits in training compared to MagicBrush, we hypothesize that this is reflected in its attention patterns. We illustrate these attention maps in Figure 3.5 of the Appendix. Compared to MagicBrush, **AURORA** pays attention to a broader area starting in the middle layers of the U-Net, possibly since movement requires "scouting" the space where placing a new object is reasonable. In the upper layers it narrows down on precise object placement. Details in Section C.11.

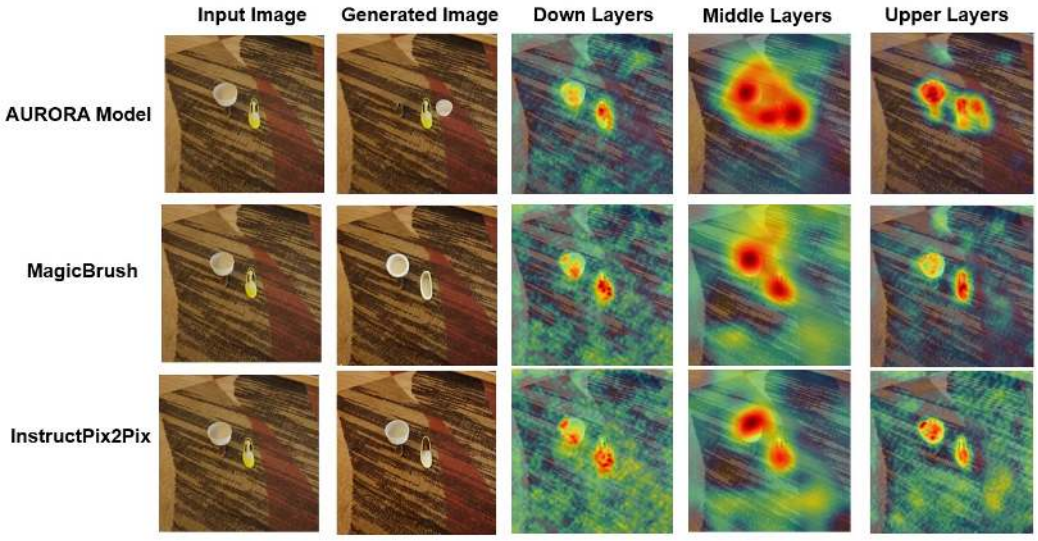


Figure 3.5: Prompt: *Put the white porcelain ramekin on the right hand of the brown shoe.* We show attention maps for three levels of U-Net layers: Down, Middle, Upper.

Common verb and nouns in AURORA vs. MS-COCO: We investigate if the distribution of verbs between broad generic VL captioning datasets and datasets tailored for action-editing differs significantly. A lot of edits have generic verbs like "move", which is nonetheless often still a complex task: "Move the cup closer to the plate" is a very different move than "Move the hand closer to their hair", where the exact action required is implicit in the scene/affordances/angles and not reflected in the textual "move". This is inherent to the editing task itself where captions tend to be shorter than traditional caption datasets, often with simpler verbs "make OBJECT ATTRIBUTE" (make the horse darker) or "replace/add OBJECT". So another interesting comparison is the distribution of verbs in traditional (object/attribute) editing vs. our focus on action editing. Finally, we note that a lot of complexity in our data comes from other linguistic constituents such as prepositions or adjectives/adverbs, e.g. "Move the hand slightly closer under the table with the finger pointing upward" where [slightly, closer, under, upward] are all interesting to understand but not verbs. To study the frequency differences to established caption datasets, we visualize the verb and noun distribution in MS-COCO, AURORA as well as the four subsets of AURORA. See Section C.12 for detailed frequencies and figures. Overall, the verbs are less diverse but as described above a lot of the complexity comes from other textual or visual aspects. On top, the verbs are quite different to COCO and notably also quite different to more established object/attribute-centric editing such as MagicBrush. Also note that while "make" is a very frequent verb, it can often be accompanied with one of the other verbs like "make the person stand up".

3.6 Conclusion and Future Work

Edits that require an understanding of real-world dynamics (e.g. actions) and reasoning are hard, especially compared to progress on more established editing subtasks. We hope that our contributions – from diverse high-quality training data with **A**U**RO**R**A**, to rigorous evaluation with **A**U**RO**R**A**-BENCH and a

new state-of-the-art model– will pave the road for further progress on building *truly general* image editing models. Understanding how to improve action and reasoning-centric editing also relates to a more fundamental problem: world modelling, i.e. predicting the next observation after taking an action on the current one. This form of image editing can be seen as one-step controllable video generation, which in turn can be used as a **generative world-simulator** (Yang et al., 2024b; Xiang et al., 2024; Zhou et al., 2024). For example, both editing or video generation can "simulate" how a visual scene would change when a robot executes an action (Yang et al., 2024b; Black et al., 2024). Though, our results show that we are still far from achieving broad world models - see Section C.2.1 for a deeper discussion on limitations – our work is a step in that direction. It has the potential to not only enable better editing tools, but also to replace narrow rule-based simulators with generative ones for "limitless" interactive training data. It is an open question for future work whether one-step editing is the right paradigm or if generating the whole trajectory from source to target image (video generation) is needed to master the edits we study in this paper.

Acknowledgements

This research was generously supported by Vanier Canada Graduate scholarship. We are also very grateful to Oscar Manas, Rabiul Awal, Vaibhav Adlakha and Marius Mosbach for their feedback and brainstorming ideas.

4

Are Diffusion Models Vision-And-Language Reasoners?

Preface

The previous Chapters 2 and 3 focused primarily on data- and analysis-centric contributions, highlighting various phenomena at the intersection of vision and language as well as failure modes of our models and evaluation practices. This chapter contributes primarily a new modeling methodology, DiffusionITM (Krojer et al., 2023a), published at NeurIPS 2023.

Continuing the thesis's *modeling* angle on deep integration from Chapter 3, but now asking whether such integration can *emerge* rather than be trained for explicitly, the central research question of this chapter directly asks how a vision model only trained to generate images can also function as an understanding model (unifying vision-and-language generation and understanding).

Our starting point was that text-conditioned image generation models around the time of publication (Rombach et al., 2022) had shown immense qualitative success using denoising diffusion processes. This led us to ask: does generating such images require understanding them? However, unlike discriminative vision-and-language models, it is non-trivial to subject these generative models to fine-grained *quantitative* evaluation of high-level phenomena such as compositionality, as opposed to, e.g., manually labeling model generations.

We address this by introducing DiffusionITM, which repurposes a diffusion model as an image-text matcher. The underlying intuition is that a model trained to predict the noise added to an image should find denoising easier when the conditioning text semantically matches the image, and harder when it does not, making the denoising error a natural matching signal. While this naive score already works well for text retrieval tasks, we further show that it needs a correction to work reliably for image retrieval, and that our proposed fine-tuning on hard negative pairs additionally improves both discriminative performance and generation quality.

Together with concurrent work at the time (Li et al., 2023a; Clark and Jaini, 2023), this study finally allowed us to *quantitatively* evaluate impressive image generation models via the same evaluation paradigm as established vision-and-language understanding models such as CLIP (Radford et al., 2021a). Across diverse benchmarks, often requiring multimodal reasoning, we find that DiffusionITM and CLIP are on par. Later studies have raised broader questions around the relation between generation and understanding (West et al., 2024), often questioning human intuitions that creation should imply understanding.

The Krojer et al. (2023) manuscript follows as originally published.

4.1 Introduction

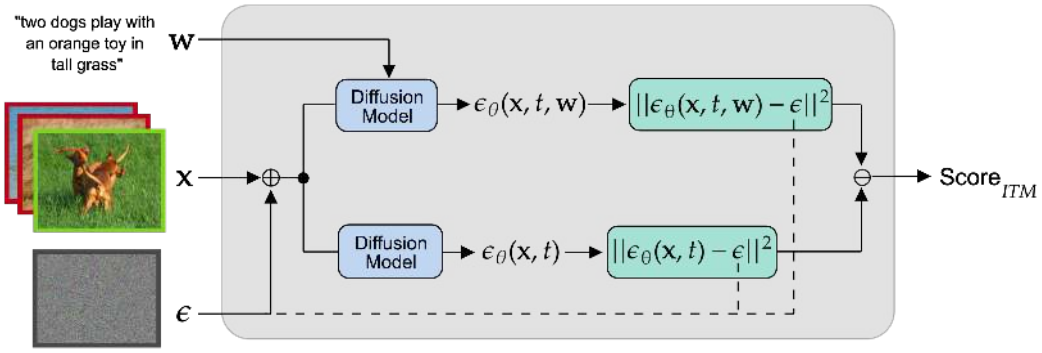


Figure 4.1: *DiffusionITM* allows us to apply a diffusion model to any image-text-matching task. It overcomes the asymmetry between image and text retrieval that previously led to random performance on image retrieval via unconditional normalization: An image is selected based on the lowest noise prediction error when conditioned on the text (upper part of figure) which is normalized by the noise prediction error without text-conditioning (lower part). With this general method the image generation community can benchmark their models on complex vision-and-language tasks such as Winoground (Thrush et al., 2022a) or ImageCoDe (Krojer et al., 2022).

Text-to-image generation is rapidly advancing. Generated images are not only highly realistic in various styles, but also reflect the compositional structure of open-ended text prompts (Chang et al., 2023; Saharia et al., 2022; Li et al., 2022b). In this work, we evaluate language-conditioned generative image models on discriminative tasks to shed light on their fine-grained understanding of vision and language. A generative objective trains a model to understand how various objects and parts compose together, and often brings non-trivial emergent capabilities with it such as latent interpolation of composite concepts (Brock et al., 2019; Rombach et al., 2022). On the other hand, discriminative vision-and-language models need only focus on the minimal information required to solve their discriminative task, which could often be spurious correlations that

don't generalize (Agrawal et al., 2016). Such erroneous understanding is then exposed downstream on image-text matching tasks purposefully designed to catch it, such as image/text retrieval using Winoground (Thrush et al., 2022a), ARO (Yuksekgonul et al., 2023b), or ImageCoDe (Krojer et al., 2022). Following these benchmarks' releases, there has been a growing focus in the vision-and-language community to fix these problems.

We hypothesize that a generative model trained to synthesize compositional data is capable of understanding the complexities required to solve hard image-text-matching tasks. To this end, we **transform a text-to-image generative model for zero-shot image-text matching**, and introduce *Diffusion Image-Text Matcher (DiffusionITM*; Figure 4.1). In this work, we use Stable Diffusion (SD) (Rombach et al., 2022) as the text-to-image model, but any other diffusion model could be used. *DiffusionITM* achieves competitive zero-shot performance on both image and text retrieval (Table 4.1).

The naive approach for image retrieval given a text prompt would be to pick the image for which SD gives the least noise prediction error, and vice versa for text retrieval. While this works well for text retrieval (Li et al., 2023a), we show it achieves random performance on image retrieval (Table 4.1). Our main insight explaining this discrepancy is that the model's success at denoising depends primarily on the visual properties of the scene, rather than equally on visuals and text (Figure 4.3). Therefore, if such a model has to select among several images given a text prompt, it will have the lowest noise prediction error for a visually familiar image regardless of the text prompt. To address this, we compute the error relative to the unconditional (i.e. no text) error (Figure 4.1). Our method outperforms existing diffusion-based discriminative methods (Li et al., 2023a; Clark and Jaini, 2023) (see Table 4.1).

Since the original generative pretraining objective of noise prediction is far from the ultimate downstream task of image-text-matching, we explore a novel discriminative finetuning scheme with hard negative image-text-pairs on MS-COCO (Lin et al., 2014). We find that this transfers well to other datasets, and

improves discriminative performance (Table 4.1) as well as generated images from DrawBench prompts (Saharia et al., 2022).

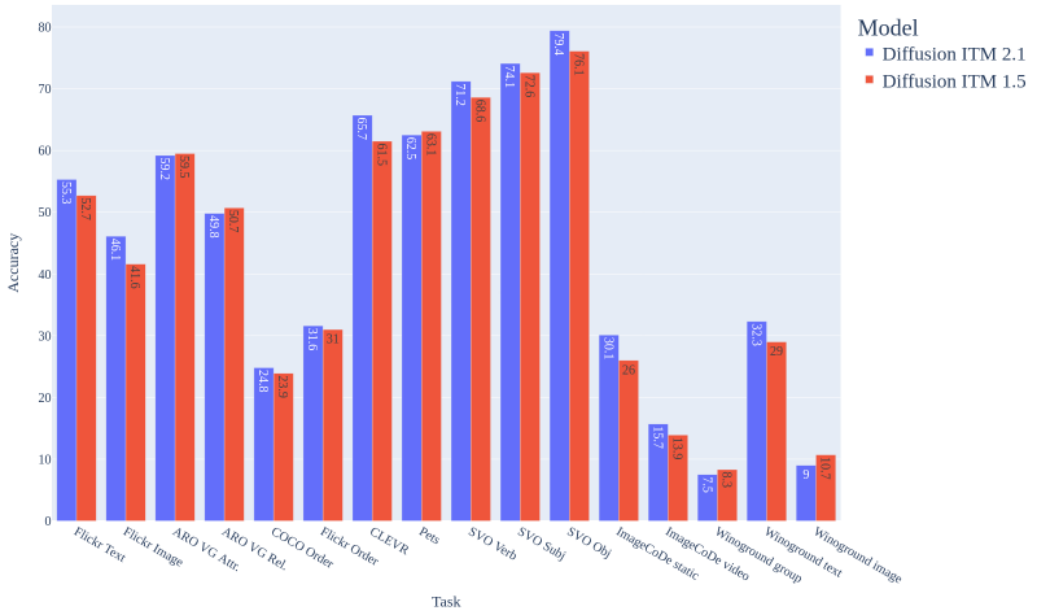


Figure 4.2: Progress of Stable Diffusion from 1.5 to 2.1 on *GDBench* tasks. *GDBench* allows fine-grained comparison of models.

Finally, we present the *GDBench* to foster research progress on image generation. Our *DiffusionITM* method enables a new automatic, fine-grained, and downstream way to evaluate diverse skills in text-conditioned image generation. Thus, once we cast an image generation model as a discriminator, we can draw from the rich history of diagnostic datasets and detailed analyses in the vision-and-language literature to analyze the model's reasoning ability. Evaluation of image generation has been notoriously difficult, and recent proposals advocate using secondary model probes on generated images to test for fine-grained ability (Hu et al., 2023; Cho et al., 2022). In contrast, we directly evaluate diffusion models on downstream tasks without requiring any probes. In the same spirit as GLUE (Wang et al., 2018), we systematically select 7 image-text-matching

tasks covering diverse reasoning skills from traditional image retrieval to diagnostic tasks (i.e. compositionality). In addition, *GDBench* also includes a bias evaluation dataset which we use for social bias analysis.

GDBench allows head-on comparison between generative models, as well as with discriminative models like CLIP (Radford et al., 2021a). Our results are as follows:

Stable Diffusion (SD) vs. CLIP: SD is competitive with CLIP on many tasks, and outperforms it on challenging compositional tasks like CLEVR and Winoground Text (Table 4.1). Further tuning of SD on MS-COCO with hard negatives outperforms vanilla SD on both image and text retrieval (Table 4.2). SD has lower stereotypical bias than CLIP (Table 4.3).

Stable Diffusion (SD) 1.5 vs 2.1: We observe varying degrees of progress, but overall SD 2.1 outperforms SD 1.5 (Figure 4.2). SD 2.1 is less biased than SD 1.5, contrary to the common trend of increased bias in larger, stronger models (Nadeem et al., 2021).

Image generation: Remarkably, improved discriminative performance via finetuning on MS-COCO also leads to improved high-level understanding of image generation. We find higher image-text-alignment on DrawBench (Saharia et al., 2022) prompts compared to vanilla SD (section D.2).

4.2 Related Work

Vision-And-Language Understanding: Widely popular tasks for vision-and-language understanding are often framed as discriminative tasks, such as image retrieval (Miech et al., 2021), VQA (Agrawal et al., 2015) or REC (Yu et al., 2016). We focus on image-text-matching (ITM) tasks (assigning a score to an image-text-pair) due to their general applicability and simplicity. Noisy image-text pairs are used for large-scale ITM training using contrastive learning (Radford et al., 2021a), and has been traditionally tackled either via dual encoders like CLIP (Radford et al., 2021a), or models with richer cross-modal interaction like BLIP (Li et al., 2022a). At the same time, ITM allows probing models in a controlled

manner with simple metrics (Thrush et al., 2022a; Hendricks and Nematzadeh, 2021b), as opposed to more complex metrics on generative tasks (Hessel et al., 2021; Saharia et al., 2022). There has been a growing interest on diagnostic benchmarks for compositionality that use hard negatives (Yuksekgonul et al., 2023b; Thrush et al., 2022a; Krojer et al., 2022; Hendricks and Nematzadeh, 2021b; Lewis et al., 2022; Parcalabescu et al., 2022). This rich literature can now be transferred to the world of image generation with our *GDBench* benchmark, enabling fine-grained analysis and benchmarking of text-conditioned image generation.

Repurposing Text-Conditioned Diffusion Models: Text-conditioned diffusion has shown remarkable advancements not only in image quality but also image-text alignment (Rombach et al., 2022; Saharia et al., 2022). A natural next question is how to leverage these abilities for other tasks (Burgert et al., 2022). Two concurrent studies use similar methods to our *diffusionITM* in a more restricted setting: *Diffusion Classifier* (Li et al., 2023a) performs image classification using SD by selecting the class with the lowest noise prediction error; (Clark and Jaini, 2023) puts more focus on timestep-weighting, and uses Imagen (Saharia et al., 2022) instead of SD. However, they only show competitive results for text retrieval, emphasizing image classification as a special case. Many vision-and-language reasoning tasks are also framed as image retrieval (Krojer et al., 2022; Hendricks and Nematzadeh, 2021b; Thrush et al., 2022a). In this work, we tackle the broader scope of vision-and-language understanding, generalize our method to image retrieval, and study hard-negative fine-tuning and transfer.

Evaluation of Image Generation: Image generation is traditionally evaluated along the two axes of image quality and image-text alignment, using metrics based on individual examples. The recently proposed TIFA metric (Hu et al., 2023) relies on an additional VQA model to answer a set of LLM-generated questions about the generated image. More traditional metrics include FID (Heusel et al., 2017) for image quality; CLIPScore (Hessel et al., 2021; Kim et al., 2022) for

image-text alignment based on CLIP-embedding; object-centric metrics (Hinz et al., 2020; Cho et al., 2022) leveraging object detectors such as DETR (Carion et al., 2020); and caption-based metrics (Hong et al., 2018) like BLEU on captions of the generated image. In contrast, *GDBench* is not a metric on individual examples, but rather a holistic evaluation framework. *GDBench* does not require another large model (e.g. VQA) for evaluation, and can be run on many diverse datasets.

Bias in Image Generation Models: It is well known that LLMs learn and amplify harmful biases present within text training corpora (Caliskan et al., 2017). Due to the lack of automatic evaluation techniques for generative models, bias investigation has mostly focused on discriminative vision-and-language (VL) models (Srinivasan and Bisk, 2022; Janghorbani and De Melo, 2023). Only few works have tackled bias in recent text-to-image models (Luccioni et al., 2023; Cho et al., 2022) and to our knowledge only (Luccioni et al., 2023) focus on bias alone: They quantify social biases by generating images over several social groups (ethnicity and gender) and measuring their variation over selected attributes (gendered adjectives and professions). They found that SD 1.4 and 2.0 are biased towards groups "associated with whiteness and masculinity" across target attributes, and that SD 2.0 was more biased than 1.4. While their methods are thorough and work for black-box systems, the evaluation is quite time consuming and manual.

4.3 Our Approach to Image-Text Matching with Diffusion Models

4.3.1 Diffusion Image-Text Matching: Overcoming the modality asymmetry for image retrieval

We present our method *Diffusion Image-Text Matching (ITM)*. Our goal is to assign a score to an image($\mathbf{x}$)-text($\mathbf{w}$) pair ($\mathbf{x}, \mathbf{w}$) which is broadly useful for downstream applications. We provide ($\mathbf{x}, \mathbf{w}$) to the diffusion model and task it to

"edit" the image according to the text. Our main intuition is if the image is not described by the text, a lot of edits are needed to fit the text, in which case it gets a low score, and vice-versa. See section D.3 for visualization.

Text-conditioned Diffusion: The objective of diffusion models is to denoise an image $\mathbf{x}$ by predicting the noise ϵ added to its clean version, conditioned on text $\mathbf{w}$ and noise level t:

$$\textit{Diffusion loss:} \quad \mathbb{E}_{\mathbf{x},\epsilon,t}\left[\|\epsilon - \epsilon_\theta(\mathbf{x}, t, \mathbf{w})\|_2^2\right] \tag{4.1}$$

Intuitively, the predicted noise is farther from the true noise when the image-text do not fit together. To transform this for ITM tasks, the sample with the lowest L2-distance of predicted and true noise could be used to select among a set of image-text-pairs. (Li et al., 2023a) and (Clark and Jaini, 2023) (concurrent works) have focused primarily on text retrieval (with classification as a special case), where the model selects from a number of texts (or class names for classification):

$$\textit{Text retrieval:} \quad \arg\min_{\mathbf{w}} \mathbb{E}_{\epsilon,t}\left[\|\epsilon - \epsilon_\theta(\mathbf{x}, t, \mathbf{w})\|_2^2\right] \tag{4.2}$$

However, naively applying this in practice would imply sampling a different ϵ for each pair $(\mathbf{x}, \mathbf{w})$ to reduce the variance in L2-distance. (Li et al., 2023a) a) sample many (hundreds!) noise-timestep pairs (ϵ, t) uniformly, and b) crucially keep the sampled (ϵ, t) constant across different $\mathbf{w}$ when calculating Equation 4.2. Finally, the guidance scale is kept at 0, thereby discarding unconditional noise prediction. This could be repurposed to perform image retrieval by iterating over images instead of text:

$$\textit{Naive image retrieval:} \quad \arg\min_{\mathbf{x}} \mathbb{E}_{\epsilon,t}\left[\|\epsilon - \epsilon_\theta(\mathbf{x}, t, \mathbf{w})\|_2^2\right] \tag{4.3}$$

We observe that while this results in SD as a strong text retriever, it achieves random chance on all image retrieval tasks! Interestingly, (Li et al., 2023a) observed that a Bayesian posterior from a discrete-label class-conditional generative

Figure 4.3: Denoising losses for two similar Flickr30K images Image1 and Image2, and Caption1 of Image1. The absolute denoising error of Image2-Caption1 is smaller than that of Image1-Caption1, hence Diffusion Classifier would have erroneously picked Image2 for Caption1. Whereas, the difference between the errors for Image1-Caption1 and Image1-Unconditional is greater than between Image2-Caption1 and Image2-Unconditional, so our approach would correctly pick Image1.

model can be approximated using the analysis:

$$p_\theta(\mathbf{c}_i \mid \mathbf{x}) = \frac{p(\mathbf{c}_i)\, p_\theta(\mathbf{x} \mid \mathbf{c}_i)}{\sum_j p(\mathbf{c}_j)\, p_\theta(\mathbf{x} \mid \mathbf{c}_j)} \approx \frac{\exp\left\{-\mathbb{E}_{t,\epsilon}\left[\|\epsilon - \epsilon_\theta(\mathbf{x}_t, \mathbf{c}_i)\|^2\right] + const.\right\}}{\sum_j \exp\left\{-\mathbb{E}_{t,\epsilon}\left[\left\|\epsilon - \epsilon_\theta(\mathbf{x}_t, \mathbf{c}_j)\right\|^2\right] + const.\right\}}. \tag{4.4}$$

(Li et al., 2023a) use pairwise samples between pairs of labels $\mathbf{c}_i$ and $\mathbf{c}_j$, to produce a low complexity approximate computation reminiscent of paired differences tests in statistics. Their analysis is based on a marginal across all possible discrete label captions $\mathbf{c}$ with respect to an image $\mathbf{x}$. In contrast, in our approach, we replace the marginal in the denominator with simply a sample from the unconditional diffusion model for $p(\mathbf{x})$, and operate in log space. The intuition behind our approach is that taking an integral over all possible caption strings ($\mathbf{w}$) is equivalent to simply the unconditional, since the "caption" dimension is (implicitly) marginalized out. Figure 4.3 illustrates another view of this intuition that for a correct text-image pair (Image1-Caption1), the denoising error is visibly smaller than for all mismatched text-image pairs for the same image (Image1). Moreover, for an incorrect but similar image (Image2), the denoising error for Caption1 is close to random, but could be less than that of Image1-Caption1 in absolute value. The incorrect Image2 would be selected regardless of the text since it is visually easier to denoise. Hence, the denoising error depends almost

entirely on the image, independent of text conditioning ("modality asymmetry").

This analysis naturally leads to the intuitive solution of normalizing the error by the unconditional (no text) error. After all, we only care about the relative difference of how much easier or harder it becomes to denoise the image with a given text relative to when no text is given (see Figure 4.1):

$$\textit{Image retrieval:} \quad \arg\min_{\mathbf{x}} \mathbb{E}_{\epsilon,t}\left[(\|\epsilon - \epsilon_\theta(\mathbf{x}, t, \mathbf{w})\|_2^2 - \|\epsilon - \epsilon_\theta(\mathbf{x}, t)\|_2^2)\right] \tag{4.5}$$

4.3.2 *HardNeg-DiffusionITM*: Tuning with compositional hard negatives and transfer

Our goal is to transform diffusion-based models for discriminative image-text-matching (ITM). However, the denoising diffusion objective only considers positive image-text pairs, and the large pre-training corpus LAION (Schuhmann et al., 2021) contains many noisy/simple examples, not conductive to complex linguistic reasoning. In contrast, models specifically dedicated to vision-and-language reasoning such as CLIP (Radford et al., 2021a) or BLIP (Li et al., 2022a) were pre-trained with negatives and, in the case of BLIP, had access to high-quality curated image-text data. Previous baselines (Li et al., 2023a) ditched the generative objective and turned it fully discriminative by finetuning a ResNet on top of the frozen mid-layer features of the U-Net for each dataset separately. We instead adopt parameter-efficient finetuning with LORA layers (Hu et al., 2022) that are added to the cross-attention from U-Net to the text, so as not to deviate too far from pretraining representations.

We address the lack of high-quality image-text-data by fine-tuning the diffusion model on MS-COCO (109K examples) with the standard diffusion objective (see Equation 4.1). As MS-COCO contains diverse high-quality image-text pairs, we finetune only once, and evaluate using *GDBench* tasks. This could be thought of as a second limited pre-training.

We address the lack of negative examples by adopting the hard negatives from (Yuksekgonul et al., 2023b) on MS-COCO: swapped text elements of the

same part-of-speech (e.g. "Men keep watch on a herd of goats" → "Goats keep watch on a herd of men"), and CLIP-based image hard negatives. The naive approach would be to minimize the noise prediction error on positive pairs $(\mathbf{x}, \mathbf{w}_{pos})$, and maximize for negative pairs $(\mathbf{x}, \mathbf{w}_{neg})$. However, if this inverse loss were applied unhinged with potentially infinite gains, it would lead to the model predicting non-sense for everything. Therefore we threshold the hard-negative error at a relative scaling factor λ of the value of the positive error:

$$\mathscr{L} = \mathscr{L}_{pos} + \text{clip}\left(\mathscr{L}_{neg}, |\lambda \mathscr{L}_{pos}|\right) \text{ where} \tag{4.6}$$

$$\mathscr{L}_{pos} = \mathbb{E}_{\mathbf{x},\epsilon,t}\left[\|\epsilon - \epsilon_\theta(\mathbf{x}, t, \mathbf{w}_{pos})\|^2\right], \quad \mathscr{L}_{neg} = -\mathbb{E}_{\mathbf{x},\epsilon,t}\left[\|\epsilon - \epsilon_\theta(\mathbf{x}, t, \mathbf{w}_{neg})\|^2\right] \tag{4.7}$$

We choose relative thresholding since we want to ensure that the model never deviates too much from its original objective of noise prediction from positive prompts. Hence $\mathscr{L}_{neg}$ is clipped between $[-\mathscr{L}_{pos}, \mathscr{L}_{pos}]$. Unlike CLIP, we cannot include a large number of hard negatives in our batches since *diffusionITM* encodes image and text together. The resulting model, *HardNeg-DiffusionITM*, is still evaluated in a zero-shot fashion on the target evaluation tasks, i.e., how well does *DiffusionITM* trained on MS-COCO transfer to target tasks.

4.4 Data: The *GDBench* Benchmark

There is a fundamental need to measure downstream performance of diffusion-based generative models on a wide range of vision-and-language reasoning tasks, to facilitate quick improvement and progress tracking. This has worked in the past, i.e. with the NLP GLUE benchmark (Wang et al., 2018). With this motivation, we introduce *GDBench*, a benchmark of eight diverse image-text-matching (ITM) tasks to explore many types of vision-and-language reasoning. These include 7 ability-centric and 1 bias dataset for the image generation community (examples shown in Figure 4.4). Additionally, most *GDBench* tasks have further fine-grained scores on sub-phenomena without requiring manual inspection, as is the case with evaluating generative models (Saharia et al., 2022). ITM as a reasoning benchmark offers simplicity and a surprising amount of diversity. After all, ITM

has become a standard paradigm for diagnostic vision-and-language datasets (see task list below) and therefore allows interpretable evaluation on many downstream skills. *GDBench* is similar to the spirit of downstream evaluation of generative models such as in TIFA evaluation (Hu et al., 2023) which makes use of a secondary model like VQA on the generated images and shows that downstream performance correlates with image quality. Whereas, we stay closer to the generative realm and evaluate generative models on several discriminative tasks without the need for secondary models. Below we introduce each dataset, asking "What phenomena does it cover that others do not?"

Flickr30K: *A lady in a blue dress riding on a scooter* (left) and CLIP-based hard negative (right)

ARO: *The man is wearing the shirt* (positive) and *The shirt is wearing the man* (hard negative)

Pets: *A photo of a english cocker spaniel*

CLEVR: *A large sphere and a small cylinder* (positive) and *A small sphere and a large cylinder* (negative)

SVO: *Person standing at window* (left) and *Person sitting at window* (right)

ImageCoDe: *There is an equal amount of yellow and white between both hands* (left) and hard negative (right)

Winoground: *Some plants surrounding a lightbulb* (left) and *A lightbulb surrounding some plants* (right)

Bias: Christian (left), Jewish (right). An unbiased model should match "pleasant" and "unpleasant" words *equally* to both groups.

Figure 4.4: Examples from the datasets in *GDBench*.

Flickr30K (Young et al., 2014) is a well-established open-ended image and text retrieval dataset, captioning diverse scenes involving people. Note that we changed the task setup to reduce computation overhead: A model has to retrieve among the 10 most similar examples based on the CLIP embedding space. While this might not be fair towards our CLIP baselines, we chose this setup primarily to study if SD could be useful a second-stage slow retriever, after a fast retriever like narrows down the candidates (Miech et al., 2021). Both **Winoground** (Thrush et al., 2022a) and **ARO** (Yuksekgonul et al., 2023b) are diagnostic benchmarks

for compositionality. Winoground is carefully curated at the cost of only 400 examples and many SOTA models have not reached significantly beyond random chance. ARO is automatically generated on top of Flickr30K, MS-COCO, and others with only compositional hard text negatives. **ImageCoDe** (Krojer et al., 2022) is an image retrieval task focusing on highly similar images with complex pragmatic captions crowdsourced from a guessing game. **SVO** (Hendricks and Nematzadeh, 2021b) disentangles performance along different parts-of-speech by pairing images that differ only in subject, object or verb. (Lewis et al., 2022) introduced a diagnostic controllable benchmark based on simple synthetic **CLEVR**[1] images of 3D shapes, thereby isolating various phenomena like attribute binding or spatial relations. Finally, we include **Pets** (Parkhi et al., 2012) as a smaller quick-to-run image classification dataset. In contrast to linguistically complex *GDBench* tasks, Pets covers the complementary skill of fine-grained recognition (37 animals).[2]

Measuring Bias: Bias evaluation is essential as large-scale models increasingly suffer from harmful biases that can impact downstream tasks. *Diffusion-ITM* allows automatic, quantitative bias evaluation with bias datasets intended for discriminative vision-and-language models. We utilize the dataset from (Janghorbani and De Melo, 2023) and investigate three different types of social biases: religious, nationality, and sexual orientation. Quantitatively, bias is present when there is a substantially stronger association of one target group to pleasant attributes compared to unpleasant attributes over another group, as measured by some notion of distance or similarity. The target groups are represented by sets of images X^I, Y^I and the attributes are captured by sets of words A^T, B^T. Bias is measured by the following normalized association score, d, called the *effect size*: [3]

[1]We generate CLEVR images and captions ourselves and they are therefore not exactly comparable with the same previous task formulation (Lewis et al., 2022; Clark and Jaini, 2023).

[2]We chose Pets over ImageNet with 1000 classes since we want *GDBench* to be light-weight to run. Following (Radford et al., 2021a), we use the text prompt "an image of X" .

[3]We use a permutation test to compute the significance of effect size scores (Caliskan et al., 2017).

Table 4.1: Benchmarking Diffusion ITM with vanilla SD and hard-negative fine-tuning on MS-COCO on *GDBench*. Diffusion Classifier performs around random chance on image retrieval.[4] Hard negative transfer finetuning significantly improves on both.

(a) Accuracy of *DiffusionITM* and baselines on *GDBench* **image retrieval tasks**.

		SVO			**ImageCoDe**		**Winoground**
	Flickr Img	Verb	Subj	Obj	Static	Video	Image
CLIP RN50x64	71.9	82.2	85.8	89.3	51.6	23.5	**13.8**
OpenCLIP ViT-L/14	**79.9**	**85.6**	**90.7**	**92.8**	**69.8**	**26.0**	11.25
Diffusion Classifier	11.1	50.6	55.0	48.7	8.9	9.0	0.25
DiffusionITM (Ours)	61.5	77.3	80.5	86.2	42.5	21.1	10.2
HardNeg-DiffusionITM (Ours)	66.1	77.8	81.3	84.7	44.9	21.0	12.3

(b) Accuracy of *DiffusionITM* and baselines on *GDBench* **text retrieval tasks**.

		ARO				**CLEVR**	**Pets**	**Winoground**
	Flickr Txt	VG Attr.	VG Rel.	COCO Ord.	Flickr Ord.	avg	**Pets**	Text
CLIP RN50x64	65.8	62.7	50.8	**52.0**	**58.8**	59.5	88.4	26.3
OpenCLIP ViT-L/14	71.3	59.2	50.3	30.8	39.8	64.3	**89.9**	30.25
Diffusion Classifier / **DiffusionITM (Ours)**	69.8	62.9	50.0	23.5	33.2	67.9	79.7	37.5
HardNeg-DiffusionITM (Ours)	**73.5**	**67.6**	**52.3**	34.4	48.6	**73.3**	81.3	**39.5**

$$d(X^I, Y^I, A^T, B^T) = \frac{\text{mean}_{x\in X}\,\psi(x,A,B)}{\text{stdev}_{i\in X\cup Y}\,\psi(i,A,B)} - \frac{\text{mean}_{y\in Y}\,\psi(y,A,B)}{\text{stdev}_{i\in X\cup Y}\,\psi(i,A,B)} \tag{4.8}$$

$$\text{where}\quad \psi(i,A,B) = \underset{a\in A}{\text{mean}}\,\sigma(i,a) - \underset{b\in B}{\text{mean}}\,\sigma(i,b)$$

and $\sigma(\cdot,\cdot)$ is our proposed *DiffusionITM* score, or, in the case of CLIP, cosine similarity.

More concretely, in the case of Religion, the image sets might be X = Christians, Y = Muslims and the attribute word sets would be A = {"joy", "trust",

[4] Random accuracy: 10% (Flickr, ImageCoDe), 50% (ARO Rel./Attr., SVO, CLEVR), 25% (Winoground), 20% (ARO Order), 2.7% (Pets).

"good", ...}, B = {"corrupt", "vulgar", "bad", ...}. Here, a positive effect size means that it is biased towards Christians and a negative effect size means it is biased towards Muslims. We provide more details under limitations in section D.1.

4.5 Experiments and Results

Our main two findings are summarized in Table 4.1: First, zero-shot *Diffusion-ITM* achieves performance near CLIP on image retrieval (Table 4.1a), overcoming the close-to-random performance of Diffusion Classifier (Li et al., 2023a). Second, our best hard negative transfer-finetuning strategy improves performance across the board, on both image and text retrieval.

Hyperparameters: Based on ablations in (Li et al., 2023a), we adopt most of their setup but aim for more simplicity: Timesteps t are sampled uniformly from $[0, 1000]$, guidance scale is kept at 0, but we drop the complicated procedure that iteratively prunes classes after a number of noise-timestep samples (ϵ, t). Instead we keep samples constant at 250 for the main zero-shot experiments in Table 4.1 and reduce it to a much more feasible number of 10 samples for other experiments.[5] This is aligned with our goal that *GDBench* should be easily adopted by researchers to evaluate their image generation method. As shown in Appendix Figure D.5, performance is not at its maximum with few samples but the trends can be studied in the same way when comparing models along different tasks. We adopt the common CLIP RN50x64 baseline and OpenCLIP ViT-L/14 for a fair comparison since SD 2.1's text backbone is from the latter. We fine-tune on the MS-COCO hard negative training set (Yuksekgonul et al., 2023b) with $lr = 1e-4$, $\lambda = 1.0$ and batchsize 112. We select a checkpoint after 8 epochs based on hard negative validation. **Runtime:** With 10 noise samples per image-text-pair evaluation on Flickr30K Text Retrieval validation takes 68 minutes on a single NVIDIA RTX A6000 GPU (compared to around 4 minutes with OpenCLIP ViT-L/14). We point to (Li et al., 2023a) for an in-depth analysis of runtime. We emphasize that we envision future stronger SD-based discriminators as slow

[5] With the exception of 20 steps for bias evaluation.

retrievers that are applied to a small set of candidates provided by a fast retriever (Miech et al., 2021), as well as the benefit of automatic evaluation.

Table 4.2: Comparison of finetuning approaches on (top) image and (bottom) text retrieval, with only 10 sampling steps due to runtime feasibility (lower performance than Table 4.1 but same trends).

(a) Image Retrieval

		SVO			**ImageCoDe**		**Winoground**
	Flickr Img	Verb	Subj	Obj	Static	Video	Image
Vanilla SD	46.1	71.2	74.1	79.4	30.1	15.7	9.0
+ MS-COCO NoNeg	48.2	71.1	74.7	76.9	29.7	16.1	10.3
+ MS-COCO $\text{RandNeg}_{\text{Txt}}$	47.7	71.5	73.8	77.5	28.3	16.0	10.7
+ MS-COCO $\text{HardNeg}_{\text{Txt}}$	47.0	71.3	74.1	76.8	30.6	16.2	9.6
+ MS-COCO $\text{HardNeg}_{\text{Img}}$	52.9	73.1	76.1	79.4	34.6	17.2	10.5
+ Hard_{Txt} + Rand_{Txt} + Hard_{Img}	49.4	71.7	75.4	78.4	31.9	16.6	9.8

(b) Text Retrieval

		ARO				**CLEVR**		**Winoground**
	Flickr Txt	VG Attr.	VG Rel.	COCO Order	Flickr Order	avg	**Pets**	Text
Vanilla SD	55.3	59.2	49.8	24.8	31.6	65.7	60.9	32.3
+ MS-COCO NoNeg	62.2	62.3	53.2	33.6	42.9	67.1	70.0	35.0
+ MS-COCO $\text{RandNeg}_{\text{Txt}}$	62.2	61.6	53.1	34.0	42.4	67.6	70.2	32.2
+ MS-COCO $\text{HardNeg}_{\text{Txt}}$	60.9	62.2	52.9	34.9	44.0	68.5	69.4	33.7
+ MS-COCO $\text{HardNeg}_{\text{Img}}$	61.1	61.1	52.4	30.6	37.6	67.4	71.0	33.7
+ Hard_{Txt} + Rand_{Txt}+ Hard_{Img}	61.7	62.0	53.1	33.9	41.2	67.0	71.0	30.8

***DiffusionITM* performance:** Our *DiffusionITM* significantly outperforms *Diffusion Classifier* on all image retrieval tasks (Table 4.1a) while CLIP still outperforms *DiffusionITM*. However on text retrieval, *DiffusionITM* even outperforms CLIP on some tasks (Table 4.1b), most notably compositionality-focused Winoground Text and CLEVR by a large margin. Here *DiffusionITM* is clearly behind CLIP on only two ARO subtasks (Flickr and COCO Order).

***HardNeg-DiffusionITM* performance:** We find that *HardNeg-DiffusionITM*

trained on MS-COCO transfers well to all tasks, outperforming *DiffusionITM* (Table 4.1). In Table 4.2 we disentangle the effect of using higher quality data and various additional hard negatives. We highlight three insights: 1) Despite fine-tuning on only a single dataset, we observe gains across almost all tasks without any negatives (*NoNeg*) explicable by the general high-quality MS-COCO dataset compared to more noisy pre-training data. 2) Using hard negatives for only one modality ($HardNeg_{Txt}$/$RandNeg_{Txt}$ vs. $HardNeg_{Img}$) only improves the respective retrieval performance while showing occasional drops in the other.[6] 3) We therefore combine all three types of hard negatives, allowing us to work with one model, *HardNeg-DiffusionITM*, rather than multiple models specific to each modality retrieval.

Table 4.3: Effect sizes for the bias evaluation. Positive effect sizes indicate bias towards target group **X**, negative effect sizes indicate bias towards **Y**. Effect sizes closer to 0 are less biased and statistically significant effect sizes at $p < 0.01$ are denoted by $*$. We see that **all models exhibit biases, with SD 2.1 being the least biased**. For brevity, Buddhist scores are omitted here but contribute to the average (details in Table D.2).

	Target **X**	Target **Y**	SD 2.1	SD 1.5	CLIP RN50x64	CLIP ViT-B/32
Religion	Christian	Muslim	**0.94***	1.06*	1.55*	1.71*
	Christian	Jewish	**1.10***	1.11*	1.54*	1.69*
	Jewish	Muslim	-0.15	0.03	0.23*	0.48*
	Hindu	Muslim	**0.86***	1.21*	1.48*	1.65*
Nationality	American	Arab	**0.63***	0.90*	0.72*	1.28*
Sexuality	Heterosexual	LGBT	**0.84***	1.04*	1.38*	1.68*
Average Absolute Effect Size			**0.65**	0.79	1.09	1.26

Stable Diffusion 1.5 vs. 2.1 Performance: 2.1 improves on the majority of *GDBench* tasks except for ARO. With *GDBench*'s diverse tasks, we can study if later versions of Stable Diffusion improve in all skill-dimensions or just some of them (see Figure 4.2). Interestingly SD 2.1 does not show significant gains

[6] Hard negatives for text are obtained by shuffling parts-of-speech, and images through CLIP image similarity.

over SD 1.5 on all of them. Most notable is ARO (compositionality): We see only minor improvements (VG tasks) or slight drops (Order tasks). At the same time, we do see a jump on Winoground Text from 29% to 32.3% and on other less adversarial tasks such Flickr30K or SVO.

Bias: Both CLIP and Stable Diffusion exhibit bias towards the dominant groups, namely Christians, Americans, and Heterosexuals (Table 4.3) with Stable Diffusion 2.1 displaying the least bias. Almost all scores are statistically significant for $p < 0.01$, with the exception of the Jewish-Muslim for both SD versions (Table 4.3) and some of the Buddhist scores (Table D.2). Version 2.1 has overall lower effect sizes (average absolute effect size of 0.65 in 2.1 vs. 0.79 in 1.5), suggesting that it is less biased than version 1.5 for this metric (Table 4.3). This goes against the trend of increased bias in stronger models (Nadeem et al., 2021). On the other hand, (Luccioni et al., 2023) found that Stable Diffusion 1.4 is less biased than 2.0. Further investigation is needed to draw a strong conclusion as to whether the 2.x versions are more or less biased than the 1.x versions due to this discrepancy. It is important to note that there was a major weakening of the safety filter between version 2.0 and 2.1, which may have affected the diversity in the training set and as such, model bias.

Analysis: We find higher image-text-alignment of images generated by *HardNeg-DiffusionITM* model based on human judgement. Although our method improves discriminative performance, does it also result in more compositional image generation? Crucially our finetuning on MS-COCO preserved the generative capabilities despite directly modifying the noise prediction. We therefore compare image-text-alignment of *DiffusionITM* against *HardNeg-DiffusionITM* on DrawBench (Saharia et al., 2022) and find promising results: From 105 complex prompts, we found that *HardNeg* is closer to the text almost twice as often as the zero-shot model. Similarly, *HardNeg* finetuning also shows slightly better image-text-alignment than *NoNeg* finetuning. For more DrawBench details see (section D.2) and other analyses (section D.6). One might ask: how complementary are the skills learned in a generative vs.

a discriminative vision-and-language model? We quantify this via the overlap of correctly predicted examples. Our hypothesis: Even if *DiffusionITM* might have lower performance on a task, its correct predictions may still cover new examples that discriminative models fail to capture. On three datasets, we compare *DiffusionITM* and two discriminative models (CLIP and BLIP) that were trained differently enough to expect varying predictions. However we find no evidence for our hypothesis and perhaps even signs of an opposite trend (Figure D.4). We speculate that this points towards the big role the text encoder behind an text-to-image model plays for vision-and-language understanding. After all, Stable Diffusion relies on a frozen CLIP text encoder.

4.6 Post-submission: The intriguing case of Stable Diffusion XL

After the paper submission, Stable Diffusion XL (SDXL) (Podell et al., 2024) was released so it was a reasonable next step to include it in our comparison of different SD versions (see Fig. 4.2). We expected SDXL to outperform previous versions, based on our upwards trend from 1.5 to 2.1 as well as findings in (Podell et al., 2024). Surprsingly, SDXL reaches significantly lower scores, below 2.1 scores on everything except three ARO subtasks, and even below 1.5 on most tasks (App. D.6 Fig. D.6 show exact numbers). Moreover, the top predictions of SDXL have the same exact scores (i.e. two images being ranked first) far more often than in SD 2.1. We confirmed that our results are not coming from a bug in our implementation and hope that future work can shed more light on quantifying the higher-level abilities of SDXL. In the meantime we offer a preliminary interpretation here: It is possible that the capabilities of image generation and image editing, specifically providing a new prompt on a partially noised image, are not always correlated. In light of this, we also offer two broader interpretations for the validity of DiffusionITM as a new evaluation methodology: Either our proposed evaluation is not generally applicable to all forms of vision-and-language skills in text-to-image models and instead measures more nuanced

aspects such as image editing or text sensitivity. Alternatively, this anomaly is evidence that our evaluation is precisely working as intended and exposes flaws that might otherwise taken longer to detect via quantitative evaluation or other metrics such as FID or TIFA.

4.7 Conclusion and Future Directions

In this paper, we introduce *DiffusionITM* and *GDBench*. *DiffusionITM* allows us to transform any diffusion-based generative model to perform image-text matching tasks. This in turn leads to an exciting new paradigm of evaluating text-to-image models on vision-and-language reasoning using diagnostic benchmarks. We also improve vision-and-language understanding of diffusion models with a novel hard negative finetuning strategy (*HardNeg*). *GDBench* allows one to study many different types of vision and language reasoning, ranging from: compositional (ARO, Winoground) and visual fine-grained reasoning (ImageCoDe), to elements of spatial/attribute binding (CLEVR). Through our experiments, we find that our proposed *DiffusionITM* approach shows vision-and-language reasoning capability through increased performance on the several evaluations, as well as provides for head-on automatic comparison among generative models. We conclude that Stable Diffusion performs competitively to CLIP, and performs better on compositional tasks. We also conclude that SD 2.1 is less biased than SD 1.5. We hope that this line of work demonstrates that high-quality diffusion methods for generating images performs well on vision-and-language reasoning. We see that the simple task of image-text-matching allows us to test many types of reasoning capabilities by carefully selecting the appropriate negatives. We encourage future research to explore this paradigm on other models families like Masked Generative Transformers (Chang et al., 2023), and with stronger backbone text encoders. While we found that improved discriminative performance translates into better image-text-alignment in generated images, this should be explored in more detail. We discuss detailed limitations (especially bias-related) in section D.1

4.8 Acknowledgements

We are grateful to the open-source community behind the Huggingface Diffusers library and the anonymous reviewers for their useful suggestions. This project was funded by the Mila-Samsung and Mila-Google grant program. SR acknowledges the support of the NSERC Discovery Grant program and the Facebook CIFAR AI Chair program. CP acknowledges the support of the Canada CIFAR AI program.

5

LatentLens: Revealing Highly Interpretable Visual Tokens in LLMs

Preface

The previous three chapters have mostly treated models as black-boxes: studying the training or evaluation data, the objective function or certain output behaviors. This final chapter, published at ICML 2026 (Krojer et al., 2026), studies model internals instead, specifically the relationship between visual and linguistic representations at different layers of LLM processing. It is also the chapter that most centrally addresses the main question of my thesis: To what extent, in what manner, are vision and language actually integrated in vision-and-language models? It thus completes the thesis's three angles by probing deep integration at the *representational* level (Chapter 1): whether vision and language come to share a common space inside the model rather than occupying

separate subspaces.

As discussed in Chapter 1, the field has witnessed various modeling paradigms for combining vision and language in a single model, ranging from early visual BERT to recent unified multimodal generation. In the previous chapters, I either focused on CLIP-based models with minimal modality interaction (Chapter 2) or diffusion-based text-to-image models (Chapters 3 and 4). In this chapter I focus on the most dominant and persistent paradigm in recent years: taking a pre-trained LLM and adapting it to understand and reason about multimodal input. Our goal is to understand why adapting an LLM to process visual input is so surprisingly easy.

A puzzling feature of the multimodal LLM paradigm is that even a fully *frozen* LLM can process visual input (in the form of visual tokens): Soon after it became clear how capable LLMs are, studies showed that it is straightforward to adapt a frozen LLM to process visual input (Lu et al., 2022; Tsimpoukelli et al., 2021; Merullo et al., 2023; Liu et al., 2023), in the simplest case all it takes is a *linear* mapping from a frozen vision encoder to a frozen LLM (Merullo et al., 2023). The intuitive hypothesis we pose in this chapter is: If the LLM is frozen and has only been trained to process *language* embeddings, the visual tokens it now receives must appear "language-like" in some way. If we take a visual token as it is processed through the LLM, does it have interpretable nearest neighbors in the LLM's representation space? Here, we define interpretability as: a visual token is interpretable if its nearest neighbors in the LLM's representation space are semantically related to its visual input. Prior work has only provided broad or largely negative answers to this question, either studying broader trends in the geometry of embedding spaces, or showing, at a small experimental scale, that naive methods to find interpretable nearest neighbors can only work on some inputs or some LLM layers (Mokady et al., 2021; Neo et al., 2025).

We address this gap by introducing LATENTLENS, a training-free interpretability method that compares visual token representations to *contextual* text representations, rather than to the LLM's static embedding or unembedding ma-

trices. Our key finding is that prior methods substantially underestimate the interpretability of visual tokens: while existing approaches (EmbeddingLens, LogitLens) label only 23–30% of visual tokens as interpretable, LATENTLENS reveals that the majority (on average 72%) are interpretable, consistently across all studied models and all LLM layers. We also identify the *Mid-Layer Leap*: even at the LLM input, visual token representations align most strongly not with static input-level embeddings but with representations from the model's middle layers, suggesting that the learned projection targets semantic rather than surface-level representations from the very start. More broadly, our findings contribute new evidence on the alignment between vision and language representations and open up new directions for analyzing the latent representations of LLMs, extending beyond VLMs to other non-linguistic tokens such as soft prompts, latent thinking tokens, or speech.

The Krojer et al. (2026) manuscript follows as originally published.

LATENTLENS Demo Code

5.1 Introduction

Transforming a large language model (LLM) into a vision-language model (VLM) can be as simple as training a linear transformation or shallow MLP that maps visual representations into the embedding space of a frozen LLM (Tsimpoukelli et al., 2021; Merullo et al., 2023; Liu et al., 2023; Beyer et al., 2024; Mañas et al., 2023, *inter alia*). While VLMs can be further improved via multiple stages of fine-tuning (Bai et al., 2023; Cho et al., 2025), the empirical success of frozen LLMs processing non-language inputs raises an important question: Why is it so easy to adapt an LLM to process data from other modalities? It has been hypothesized that LLMs are "universal computation engines" that can process arbitrary modalities with minimal adaptation (Lu et al., 2022; Shen et al., 2023; García-de Herreros et al., 2024). Through training on vast amounts of natural language, LLMs may implicitly learn about the physical world, form priors about

visual properties (Han et al., 2025), or even learn a more sophisticated world model (Patel and Pavlick, 2022). It has also been argued that separately trained vision and language representations may converge to a shared structure (Huh et al., 2024), which could facilitate training simple projections between them.

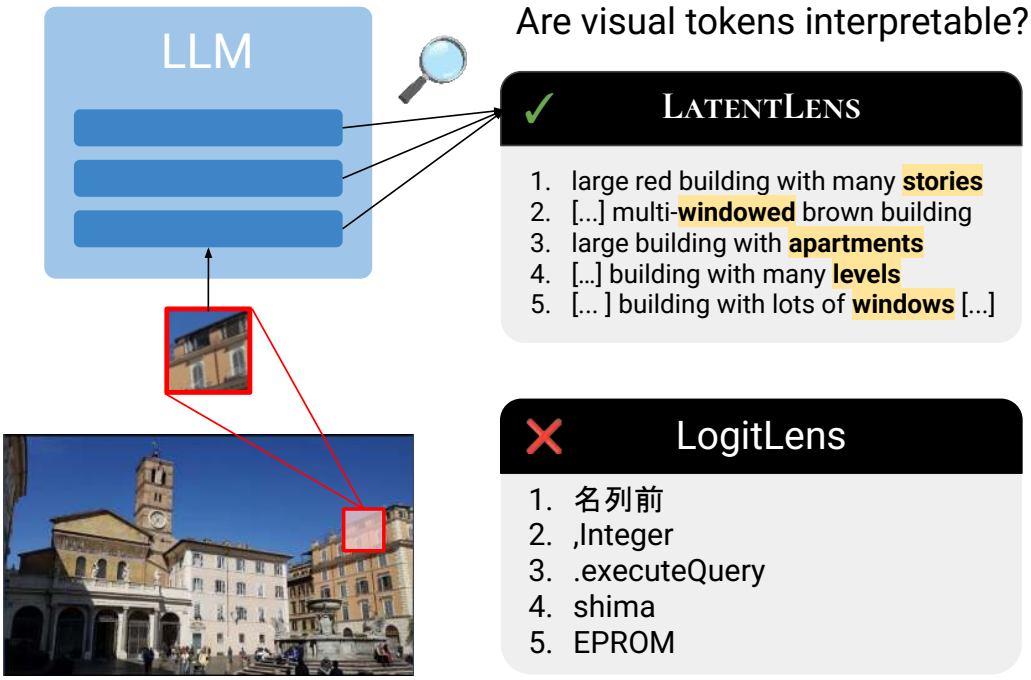


Figure 5.1: Illustration of our method. LATENTLENS compares latent representations of visual tokens to contextualized text representations obtained from full sentence descriptions.

However, these hypotheses do not explain how visual representations are integrated inside an LLM and its representation space. Are the visual tokens processed by an LLM interpretable, i.e., do their representations correspond to semantically meaningful language? Existing work suggests that visual tokens are rarely interpretable at the input level via nearest neighbors in the language model embedding space (Mokady et al., 2021; Neo et al., 2025, EmbeddingLens). Recent work has explored the utility of LogitLens, which uses the LLM unem-

bedding matrix (nostalgebraist, 2020), to interpret and analyze visual token representations (Neo et al., 2025; Jiang et al., 2025b). Finally, training-based methods such as sparse autoencoders (Cunningham et al., 2023; Venhoff et al., 2025) or supervised probing (Fu et al., 2025) have also been applied to understand what visual representations encode. However, both embedding-space and training-based methods are inconclusive about the interpretability of visual tokens in LLMs.

In this work, we propose LATENTLENS, a novel interpretability method for analyzing latent representations in VLMs. LATENTLENS is training-free and provides fine-grained sentence-level descriptions of latent representations. **Our key insight is that the most natural comparison for visual token representations are *contextual* text representations**, and not the LLM embedding or unembedding matrix. Thus, LATENTLENS compares visual token representations to their nearest neighbors in a large pool of contextualized token representations from intermediate LLM layers. For example, in Figure 5.1, the visual token of a building yields the nearest neighbor **stories** in the context of "... building with many **stories** ".

Empirically, we analyze the interpretability of visual token representations across layers for 15 different VLMs. We find that compared to other training-free approaches (LogitLens and EmbeddingLens), LATENTLENS reveals consistently high interpretability, highlighting that previous methods substantially underestimate the interpretability of visual token representations. Averaged across all 15 VLMs and all layers we study, LATENTLENS renders 68% of visual tokens interpretable (according to a VLM-judge). When using EmbeddingLens and LogitLens, however, only 32% and 24% of visual token representations are labeled as interpretable, respectively. Through ablation studies, we provide additional insights about the nature of this interpretable alignment, showing, e.g., that even linear projections lead to interpretable visual token representations. We also find evidence for a *Mid-Layer Leap* in the learned projection: the visual token representations at the input and early layers align most strongly to contextu-

alized representations from mid-layers (e.g., layers 8–16), suggesting that the learned projection targets semantic rather than lexical representations. We also present qualitative analyses showing that LATENTLENS produces rich sentence-level descriptions, unlike LogitLens which might return subwords or next-token predictions.

Overall, our findings challenge existing assumptions about the interpretability of visual tokens, and offer new insights about the alignment of vision and language representations. We encourage the reader to explore the interactive demo as well as our pip-package with easy access to our database of contextual embeddings to facilitate adoption.

5.2 Background

We first introduce technical background on VLMs and describe prior work on analyzing their latent representations.

5.2.1 Turning LLMs into VLMs

A common approach for converting an LLM into a VLM is to project representations produced by a vision encoder into the embedding space of the LLM via a learned connector. More formally, let $\texttt{venc}$ be a pre-trained vision encoder, which produces a sequence of *image embeddings* $\texttt{venc}(x_{\text{img}}) = [\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_{T_v}]$ with $\mathbf{v}_i \in \mathbb{R}^{d_v}$ for a given image x_{img}, let $\texttt{llm}$ be a pre-trained language model, and let x_{text} be a textual input. A multimodal input is constructed as follows: First, x_{text} is processed by a tokenizer $\texttt{tok}$ and then converted into token embeddings via an embedding matrix $\mathbf{E}_{\texttt{emb}} \in \mathbb{R}^{|\mathcal{V}| \times d}$. Second, *visual tokens* representing x_{img} are obtained by projecting each image embedding $\mathbf{v}_i$ into the embedding space of $\texttt{llm}$ using a projection function[1] $\texttt{proj} : \mathbb{R}^{d_v} \mapsto \mathbb{R}^d$. Finally, the visual and textual representations are concatenated $\mathbf{x} = [\mathbf{p}_1, \ldots, \mathbf{p}_{T_v}, \mathbf{e}_1, \ldots, \mathbf{e}_{T_t}]$, where $\mathbf{p}_i = \texttt{proj}(\mathbf{v}_i) \in \mathbb{R}^d \ \forall \ i = 1, \ldots, T_v$ and $\mathbf{e}_1, \ldots, \mathbf{e}_{T_t} = \texttt{emb}(\texttt{tok}(x_{\text{text}})) \in \mathbb{R}^{T_t \times d}$. The resulting multimodal sequence $\mathbf{x}$ is then processed by the $\texttt{llm}$ and converted into

[1] In practice, $\texttt{proj}$ is usually a linear layer, an MLP, or an attention-based module (Liu et al., 2023; Wang et al., 2024a).

latent representations $\mathbf{h}_i^{(\ell)} \in \mathbb{R}^d$, where i denotes the position in the sequence and ℓ a layer, respectively.[2] During the forward pass, textual representations can self-attend to the information encoded in the visual representations $\mathbf{h}_i^{(\ell)}$.

Training. Given a dataset $\mathcal{D} = \{(x_{\text{img}}, x_{\text{text}})_i\}_{i=1}^N$ of image-caption pairs, the `llm` can be trained by minimizing the cross-entropy loss over the target caption tokens:

$$\mathcal{L}_{\text{LM}} = -\sum_{t=T_{\text{instr}}+1}^{T} \log p(y_t \mid y_{<t}, x_{\text{img}}),$$

where: $\texttt{tok}(x_{\text{text}}) = [\underbrace{y_1, \ldots, y_{T_{\text{instr}}}}_{\text{instruction}}, \underbrace{y_{T_{\text{instr}}+1}, \ldots, y_T}_{\text{caption}}]$, and, unless stated otherwise, only the weights of `proj` are trained, with the `llm` and `venc` weights frozen.

5.2.2 Interpreting VLM representations

An important open question is what exactly is encoded in visual token representations as they are processed by an LLM, and to what extent are the latent visual token representations ($\mathbf{h}_i^{(\ell)}$) interpretable as language-like tokens? We will briefly introduce popular approaches for studying these questions. We note that other supervised or prompting-based approaches exist such as probing (Belinkov, 2022), SAEs (Cunningham et al., 2023), LatentQA (Pan et al., 2024) or Patchscopes (Ghandeharioun et al., 2024). Instead, we focus on methods which are training-free and directly leverage the LLMs representation space.

EmbeddingLens. The simplest approach to interpret visual token representations is to compare them to the elements of the embedding matrix $\mathbf{E}_{\texttt{emb}}$ of `llm`. This is motivated by the projection layer being trained to output representations that are compatible with the embedding space of `llm`. For this approach, each $\mathbf{p}_i$ (or its corresponding latent $\mathbf{h}_i^{(\ell)}$ at layer ℓ) is compared to the embeddings in `llm`'s embedding matrix $\mathbf{E}_{\texttt{emb}}$ via cosine similarity and the top-k most similar embeddings serve as "labels." EmbeddingLens has been previously used for interpreting soft prompts (Lester et al., 2021), visual tokens (Mokady et al., 2021; Jiang et al., 2025a) or speech (Ògúnrèmí et al., 2025).

[2] Color-coding is used for visual token representations ($\mathbf{h}_i^{(\ell)}$).

LogitLens. A slightly different but conceptually very similar approach is the LogitLens (nostalgebraist, 2020). Instead of comparing latent representations directly to embeddings, LogitLens projects the latent representation to the unembedding space of the model by multiplying $\mathbf{h}_i^{(\ell)}$ with the unembedding matrix $\mathbf{W}_{\texttt{unemb}} \in \mathbb{R}^{d \times |\mathcal{V}|}$, obtaining a distribution over vocabulary items. Then, the top-k vocabulary items, i.e., those with the largest logits, are retrieved as "labels". LogitLens been widely used in previous work analyzing LLMs (Geva et al., 2022, 2023; Fierro et al., 2025), and has recently been adopted for VLMs (Shukor and Cord, 2024; Neo et al., 2025; Jiang et al., 2025b).

5.3 LATENTLENS

In the following, we provide a unifying perspective on existing training-free methods for interpreting latent representations and introduce LATENTLENS, a novel method for mapping latent representations to natural language descriptions.

5.3.1 Unifying existing lenses

EmbeddingLens and LogitLens share the same goal and setup: To map a latent representation $\mathbf{h}_i^{(\ell)} \in \mathbb{R}^d$ at position i and layer ℓ, to a natural language description. Here, we provide a unified perspective on both methods. Formally, let $\mathcal{C}$ be a set of candidate descriptions, where each description $d_j \in \mathcal{C}$ is associated with a vector $\mathbf{r}_j \in \mathbb{R}^d$. $\mathbf{h}_i^{(\ell)}$ is mapped to a description d_j in three steps:

① **Compute scores:** For each $\mathbf{r}_j$, compute a score using a similarity function $f : \mathbb{R}^d \times \mathbb{R}^d \mapsto \mathbb{R}$:

$$s_j = f(\mathbf{h}_i^{\ell}, \mathbf{r}_j), \quad j = 1, \ldots, |\mathcal{C}|.$$

② **Select top-k vectors:** $I_{i,\ell} = \arg\text{top-}k(\{s_j\}_{j=1}^{|\mathcal{C}|})$.

③ **Return descriptions:** $\{d_j : j \in I_{i,\ell}\}$.

The set of possible descriptions for EmbeddingLens and LogitLens is given by the language model vocabulary, i.e., $\mathcal{C} = \mathcal{V}$. The similarity function is either cosine similarity with the rows of the embedding matrix $\mathbf{W}_{\texttt{emb}}$ (EmbeddingLens)

or the inner product with the output embedding matrix $\mathbf{W}_{\texttt{unemb}}$ (LogitLens). Finally, in both cases, the vocabulary items are sorted based on their similarity score and the top-k elements are selected as a description of $\mathbf{h}_i^{(\ell)}$.[3]

Limitations of prior lenses. Under our unified framework, two limitations of EmbeddingLens and LogitLens become apparent: 1) The description set is limited to a model vocabulary $\mathcal{V}$, containing only (sub-word) tokens. 2) Latent representations $\mathbf{h}_i^{(\ell)}$ from different layers are always compared to the same reference vectors, which are either the input or output embeddings. However, it is unclear whether the output or input embedding space is the most natural representation space to compare against. For example, LogitLens tends to works best for later layers, closer to the models output embedding space, and reliability can vary significantly across models (Geva et al., 2021; Belrose et al., 2023).

5.3.2 Method

We propose LATENTLENS, a novel interpretability method for mapping latent representations to descriptions in natural language. The key idea of LATENTLENS is that **the natural point of comparison for latent representations are other contextualized LLM representations** to serve as potential nearest neighbors, i.e., a token in the *context* of a sentence. Additionally, we believe that limiting the set of descriptions to individual sub-word tokens is unnecessarily restrictive and instead propose to use a large corpus of multiple-token sequences, e.g., sentences, which provides semantically richer descriptions for interpretation.

Concretely, LATENTLENS works as follows (see also Figure 5.2): Let $\mathscr{C}$ be a (large) corpus of sentences and $\mathscr{M}$ be an LLM with L layers. We construct a set of reference vectors $\mathscr{R}$ by encoding every sequence $d_j \in \mathscr{C}$ with $\mathscr{M}$ and storing the contextualized token representations $\mathbf{r}_{j,t}^{(\ell)}$ at every position t of the sequence and every layer ℓ of the model. To analyze the latent representation $\mathbf{h}_i^{(\ell')}$ of a visual token at position i and layer ℓ', we compute the cosine similarity between $\mathbf{h}_i^{(\ell')}$

[3] We note that both methods can be seen as unsupervised version of *linear probing* (Belinkov, 2022), where instead of learning a linear transformation $\mathbf{W} \in \mathbb{R}^{d \times k}$ using supervised data, one relies on the pre-trained embeddding or unembedding matrix.

and every $\mathbf{r}_{j,t}^{(\ell)} \in \mathscr{R}$, obtain the top-$k$ reference vectors with the highest similarity, and return their corresponding sequences as descriptions. We note that, in contrast to EmbeddingLens and LogitLens, LATENTLENS uses several reference vectors per description, i.e., one for each token of the description and layer of the model. Therefore ℓ and ℓ' are not necessarily equal and a top-k reference vector can come from a *different* layer than the visual token under inspection.

Following the example in Figure 5.2, a visual token representation $\mathbf{h}_i^{(\ell')}$ may have a nearest neighbor $\mathbf{r}_{j,t}^{(\ell)}$, of the contextualized representation of the token **clocks** in the sentence: "stone tower with gold **clocks**". Compared to EmbeddingLens and LogitLens, LATENTLENS can naturally be applied at every layer of a model and provides full sentence descriptions, allowing for more fine-grained interpretations.[4]

5.3.3 Evaluating interpretability

Determining whether a visual token representation is interpretable requires careful consideration of what is a semantic match between the description provided by an interpretability method and the underlying image patch. This is a complex task, where semantic matches can take many different surface forms.

We use GPT-5 (Hurst et al., 2024) as a judge to automate this process. Given an image with a red bounding box highlighting the visual token region (cf. Figure 5.1) plus the 8 surrounding visual tokens, as well as the top-5 descriptions returned by either EmbeddingLens, LogitLens, or LATENTLENS, we prompt the judge to determine whether a description is interpretable, and to classify such cases as *concrete* (directly visible), *abstract* (conceptually related), or *global* (present elsewhere in image). A visual token is *interpretable* if at least one of the top-5 descriptions is classified as interpretable. We, the authors, validate this judge against human annotations across all three methods (EmbeddingLens, LATENTLENS, and LogitLens), totaling 1,020 instances. We find substantial agreement between the LLM judge and humans with Cohen's $\kappa = 0.68$. Judge prompt

[4]While we focus on sentence-level, our approach can be trivially extended to phrases or even paragraphs.

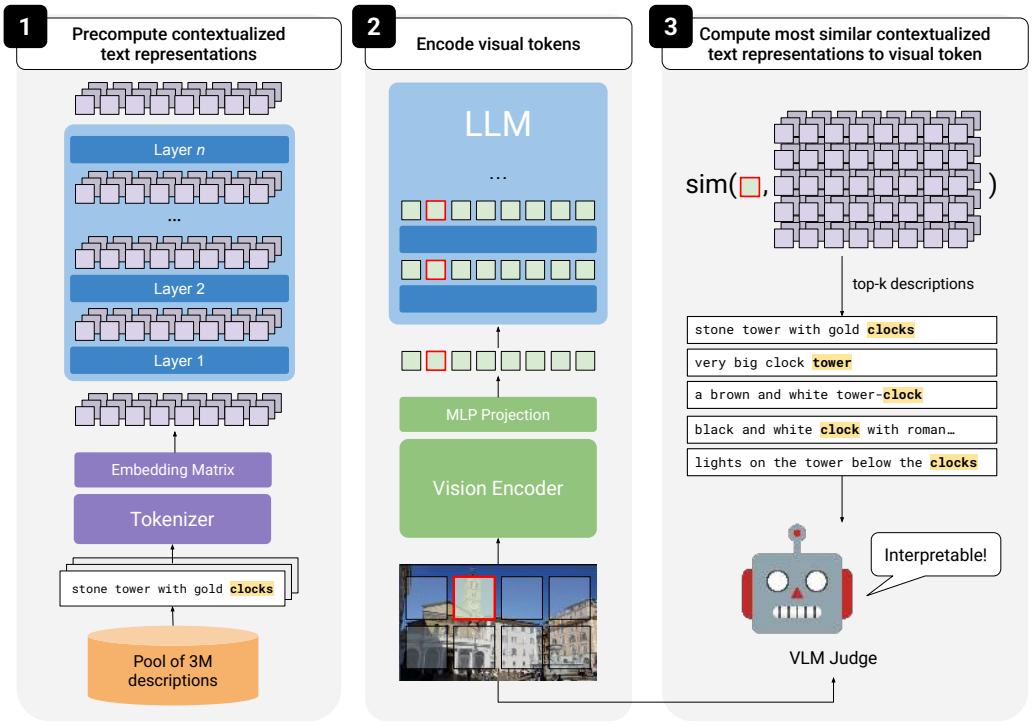


Figure 5.2: Illustration of LATENTLENS. (1) Contextualized token representations are precomputed in multiple layers of an LLM using a large corpus of descriptions. (2) Latent representations of visual tokens are extracted from all layers of the LLM, and (3) compared against the precomputed contextualized token representations. The interpretability of the visual token based on its top-k descriptions can be automatically evaluated by a VLM-judge.

and details on human annotation are in Section E.3.

5.4 Experiments

Next, we describe our controlled setup of training projections between different vision encoders and LLMs (Section 5.4.1), interpret visual tokens with LATENTLENS first in our controlled setup (Section 5.4.2) and afterwards on off-the-shelf VLMs (Section 5.4.4). We also study to which LLM layers visual tokens align to (Section 5.4.3) and ablate how the size of the pool of contextual embeddings (Section 5.4.5).

5.4.1 Experimental setup

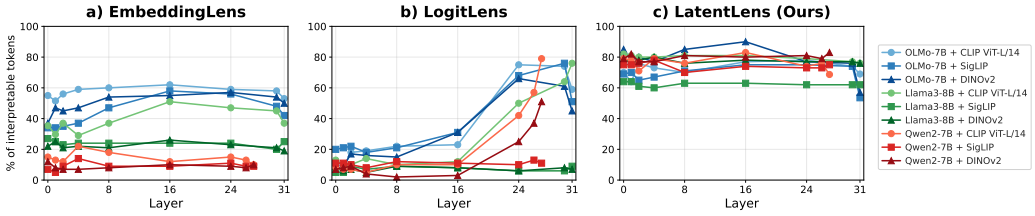


Figure 5.3: Interpretability of visual tokens across layers using three different "lenses". Each curve shows the percentage of interpretable visual tokens per layer across model. **(a)** EmbeddingLens: a large number of visual tokens is interpretable for OLMo variants but less for Llama3 and Qwen2. **(b)** LogitLens: low interpretability at early layers with a stark increase at later layers for most models. **(c)** LATENTLENS: the majority of visual tokens are interpretable across all models and layers.

Models and training. Unless stated otherwise, we train the projection function between different combinations of vision encoders and LLMs using a controlled setup. We use three LLMs (OLMo-7B (Groeneveld et al., 2024), Qwen2-7B (Yang et al., 2024a), LLaMA3-8B (Grattafiori et al., 2024)), and three vision encoders (CLIP-ViT-L/14-336 (Radford et al., 2021c), DINOv2-L-336 (Oquab et al., 2023), and SigLIP-so400M-patch14-384 (Zhai et al., 2023)), resulting in a total of 9 model combinations. We note that compared to the other vision encoders, DINOv2-L-336 was pre-trained without any textual supervision. Models are trained following Molmo using the PixMo-Cap dataset (Deitke et al., 2025),

with captions averaging 167 words and 9 sentences each. The trained projector `proj` is a 3-layer MLP and all other weights remain frozen. To reduce confounding factors, we directly use the patchified image as the input to each model. All models are trained for 12K steps with an effective batchsize of 8.

Captioning performance. We verify that our models produce reasonable captions using DCScore (Ye et al., 2025), a GPT-4o-based judge that rates captions on a 1–10 scale across fine-grained criteria (faithfulness, detail accuracy, hallucinations, completeness). Our models average 6.0/10, with CLIP and SigLIP encoders reaching an average of 6.8 while DINOv2 models score lower at avg. 4.4, likely due to DINOv2's lack of language supervision during pre-training. For reference, off-the-shelf Qwen2-VL-7B-Instruct (Wang et al., 2024a) achieves a score of 8.5/10. See Section E.13.

LATENTLENS setup. We use 2.99M Visual Genome captions (Krishna et al., 2017) as our corpus $\mathscr{C}$ of descriptions. All captions are encoded by each individual LLM, and we store all contextual token representations for layers $\ell \in \{1, 2, 4, 8, 16, 24, L\text{-}2, L\text{-}1\}$ as reference vectors.[5] This encoding is a one-time cost per LLM backbone: the forward pass through ~3M sentences takes ~2h of GPU compute, with ~13h total wall time including index building and ~26 GB of storage across 8 layers (float8). Once the index is loaded, nearest-neighbor search takes ~29 ms per image. We show in Section 5.4.5 that using just 1% of the corpus yields comparably strong interpretability while reducing storage to ~250 MB.

LLM-judge setup. Although LATENTLENS retrieves full sentence descriptions, we only provide the *full* words corresponding to the top-5 contextualized token representations as descriptions for the LLM judge. We make this decision for two reasons: 1) We found that, in contrast to human annotators, the LLM-judge can get distracted by the sentence-level context, giving inconsistent results. 2) This setup provides a fair comparison to EmbeddingLens and LogitLens (both

[5] Due to memory constraints we store at most 20 different contextual representations per token in the model's vocabulary.

only return individual tokens as descriptions) by relying on the same exact judge prompt across all methods. We note that for LATENTLENS, this approach may even underestimate interpretability and we further investigate the benefit of sentence-level descriptions in Section 5.5.

5.4.2 Visual token representations are consistently interpretable across layers

The main question we investigate is how interpretable are visual token representations as they are processed by LLMs? To answer this question, we randomly sample 100 image patches from 100 images[6] from the PixMo-Cap validation set and compare the fraction of interpretable representations for LATENTLENS, EmbeddingLens, and LogitLens[7] using the LLM judge described in §5.4.1.

Results. Figure 5.3 shows the fraction of interpretable visual token representations according to LATENTLENS, EmbeddingLens, and LogitLens for all models and layers. For EmbeddingLens (5.3a), we find that a reasonably large number of visual tokens are interpretable for some models, e.g., for all OLMo-based variants, 34–62% of visual tokens are interpretable from the input layer onward. For Qwen2-based models, however, only a small fraction of visual tokens are interpretable across layers (less than 20%). Llama3-based models fall in between with 20–50% of the visual tokens being labelled as interpretable.

For LogitLens (5.3b), we find that for all models, less than 25% of visual token representations are labelled as interpretable at lower layers. For all models except Llama3 + DINOv2, Llama3 + SigLIP, and Qwen2 + SigLIP much more visual token representations are interpretable at the later layers, e.g., 60–80% for all OLMo-based models at layers 24 and 30. This is consistent with the limitation of LogitLens discussed in Section 5.3, in particular its applicability only for layers close to the output layer. We additionally compare to Tuned Lens (Belrose et al.,

[6]To lower API costs, we test 100 patches, resulting in 3 methods × 100 patches × 9 models × 9 layers = 24.3K calls.

[7]In Section E.5 we also study an improved version of LogitLens, TunedLens (Belrose et al., 2023), and find that it does not change our LogitLens results.

2023), which learns per-layer affine probes on top of the LogitLens decoding step and find it does not improve visual token interpretability (see Section E.5 for details).

In contrast, using LATENTLENS (5.3c) results in *consistently higher interpretability scores* across all layers. Moreover, there are only minor differences in interpretability scores across model combinations: almost all models have interpretable visual tokens in the range of 60–85% across all layers, with some OLMo variants dropping to around 53% at the final layer. We perform a series of ablations (see Section E.4 for details), confirming that our findings are not tied to our specific training setup. We find no substantial change in LATENTLENS interpretability when reducing the expressivity of the mapping to linear, or when training on much shorter captions.

Overall, our results indicate that previous methods underestimate the interpretability of visual tokens and demonstrate the importance of using the right lens for analyzing latent representations. It is particularly interesting that visual tokens from DINOv2, with no explicit linguistic supervision, show consistently high interpretability with all three lenses.[8] Returning to our research question of whether visual tokens appear like interpretable words to the LLM, we can now answer: While visual token representations are not necessarily mapped one-to-one to the vocabulary of the LLM, they are often similar to contextual token representations which are semantically related to the image contents.

5.4.3 Mid-Layer Leap: Visual token representations tend to align to later layer text representations

LATENTLENS allows us to compare visual token representations at layer ℓ to contextual token representations from *any* LLM layer. We now investigate which latent textual representations are most similar to visual token representations. Given a visual token representation at layer ℓ, we obtain the top-5 contextualized

[8]The ability of DINOv2 to predict linguistic attributes of visual concepts has recently been reported (Oneata et al., 2025).

token representations with the highest cosine similarity and report the layer these contextualized representations are obtained from.

Results. Figure 5.4 shows the results of this experiment for all 9 model com-

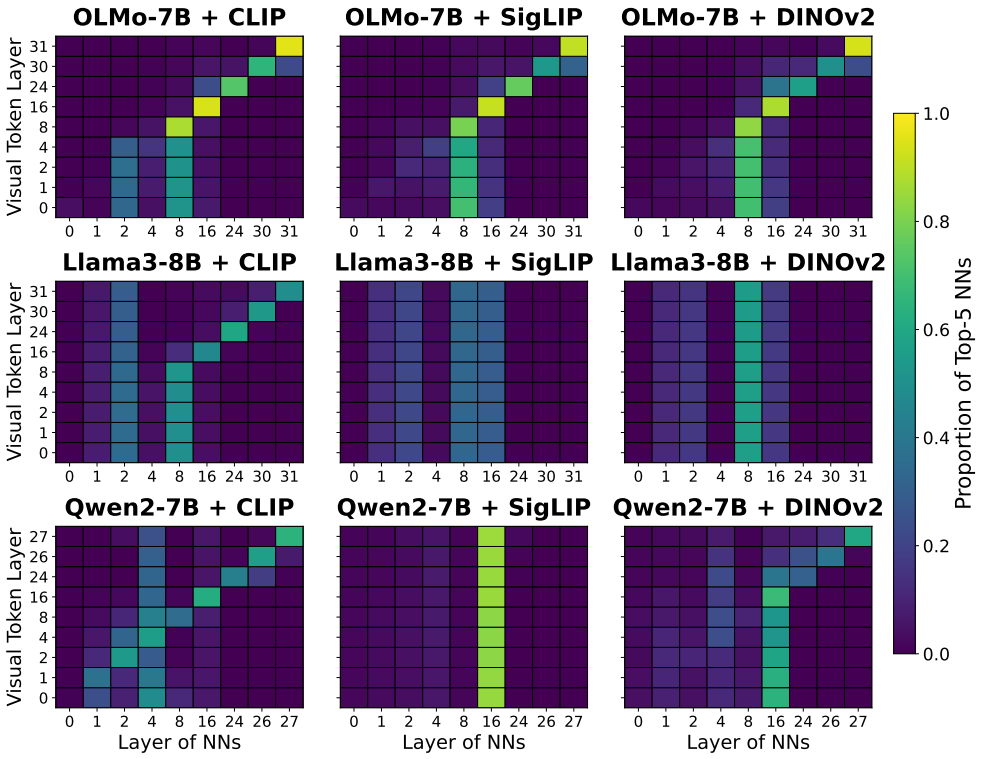


Figure 5.4: The Mid-Layer Leap: early visual tokens align to later LLM layers. For visual tokens at different stages of LLM processing, we compute their top-5 Nearest Neighbors from all other LLM layers. We find that early visual tokens, even at the input itself, align most to middle layers, e.g., layer 8 or 16. Some model combinations align most to a constant layer throughout processing, such as LLaMA3 variants. We analyze the L2 norm distributions and potential outlier effects in Section E.6.

binations. Surprisingly, visual token representations from early layers, and even the input layer, tend to have higher cosine similarity not with contextualized token representations from the same layer but instead with those from later layers. For example, with OLMo-7B + SigLIP, for visual token representations at layer 0,

the majority of the nearest neighbor contextualized representations are from layer 8 of the LLM. Only once we reach mid-layers such as layer 8, we see a diagonal pattern where visual token representations are closest to contextualized representations from the same layer. For other model combinations, results are even more surprising. For Qwen2-7B + SigLIP, the most similar representations for visual tokens from any layer are always from layer 16.

Overall, these results suggests that the visual token representations align the most with more contextualized LLM representations rather than lexical representations at the input level. We refer to this as the *Mid-Layer Leap* phenomenon and provide additional analyses of this finding in Sections E.6 and E.8, investigating how much visual token representations change across layers and whether rogue dimensions (Timkey and van Schijndel, 2021) dominate the cosine similarity. We find that visual token representations change very little throughout layers, and no systematic evidence for rogue dimensions, respectively.

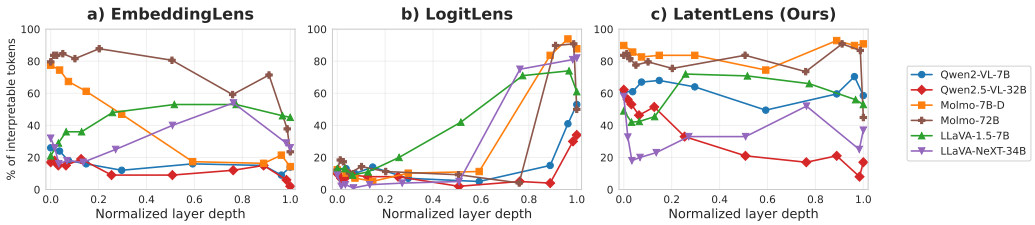


Figure 5.5: Interpretability of visual tokens across six off-the-shelf VLMs (normalized layer depth). We apply LATENTLENS and baselines to six off-the-shelf models spanning different architectures and scales. The x-axis shows normalized layer depth (0 = input, 1 = final layer) to enable direct comparison across models. LATENTLENS outperforms the baselines on all six models.

5.4.4 Results generalize to off-the-shelf VLMs

Finally, we validate that LATENTLENS generalizes beyond our controlled setup by applying all three lenses to six off-the-shelf VLMs spanning different scales: Molmo-7B-D and Molmo-72B (Deitke et al., 2025), LLaVA-1.5-7B (Liu et al., 2024a), LLaVA-NeXT-34B (Liu et al., 2024b), Qwen2-VL-7B-Instruct (Wang et al.,

2024a), and Qwen2.5-VL-32B-Instruct (Bai et al., 2025).

Results. Figure 5.5 shows interpretability across layers for all six models.

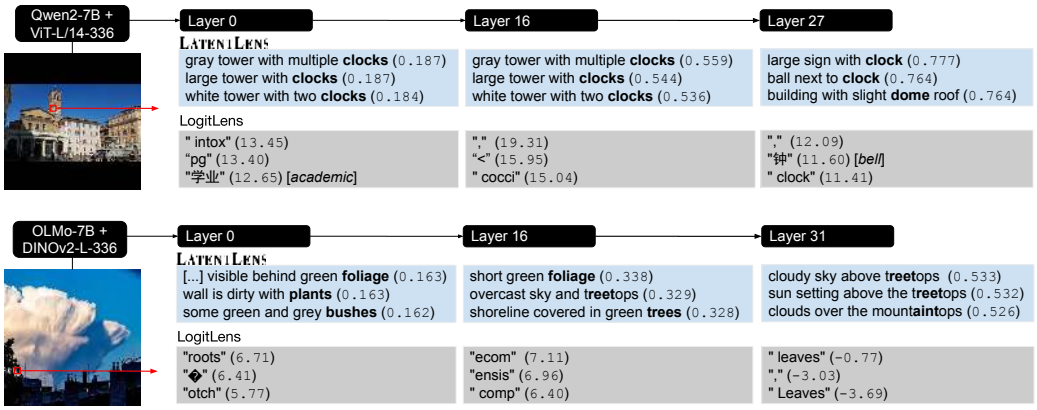


Figure 5.6: Qualitative examples. Highest-scoring descriptions are extracted from different combinations of LLMs and Vision Encoders using both LATENTLENS and LogitLens. The described patch is shown with a red outline, and the magnitude of scores are shown in parentheses. For LATENTLENS, the contextualized token used for the similarity calculation is shown in **bold**. Best viewed in colour.

LATENTLENS consistently achieves the highest interpretability on all six models. The Molmo models perform best (Molmo-7B-D: 86%, Molmo-72B: 78% average), consistent with their OLMo backbone being closest to our controlled setup. Qwen2-VL-7B-Instruct and LLaVA-1.5-7B reach 55–62% on average, while the larger Qwen2.5-VL-32B-Instruct and LLaVA-NeXT-34B are lower at 33–35%[9], though still substantially outperforming the baselines when averaged across all layers. EmbeddingLens and LogitLens typically reach 11–35% average interpretability, with the exception of EmbeddingLens on Molmo-72B (70%), also likely due to the OLMo backbone. We also verify that all six off-the-shelf models exhibit the same *Mid-Layer Leap* phenomenon observed in the controlled setup; see Section E.9 for heatmaps.

[9]It will be interesting for future work to explore why larger models have less consistent trends across layers

5.4.5 Corpus size sensitivity

A natural question is how sensitive LATENTLENS is to the size of the corpus used to build the contextual embedding index. We ablate this by randomly sub-sampling from the full contextual embedding pool at rates of 0.1%, 1% and 10%. This is done across three model pairs (OLMo+CLIP, LLaMA3+SigLIP, Qwen2+DINOv2), measuring average interpretability across representative layers (0, 8, 16, 31). As shown in Table 5.1, interpretability drops meaningfully only below 1% (~58% avg.), while the difference between 1% and the full corpus is small in practice (67% vs. 72% on average). Intuitively, a smaller corpus does not change *whether* a token is interpretable but it reduces the precision and granularity of the descriptions, since fewer sentence contexts are available to match against.

Table 5.1: Corpus size sensitivity. Average % of interpretable visual tokens (LATENTLENS) for three model pairs and layers 0, 8, 16, 31. The difference between 1% and the full corpus is small.

Corpus size	Avg. % interpretable
0.1%	58.4
1%	67.3
10%	72.5
100%	71.6

5.5 Qualitative results

The previous section showed that under the right lens, visual token representations are highly interpretable. Here, we provide qualitative examples for the interpretability provided by LATENTLENS, and compare to LogitLens.

LATENTLENS vs. LogitLens descriptions. Figure 5.6 shows qualitative examples of the highest-scoring descriptions for two image patches extracted using LATENTLENS and LogitLens (see Section E.14 and our interactive demo for additional examples). LATENTLENS descriptions are semantically more mean-

ingful than those of LogitLens. For example, even when LogitLens yields some interpretable tokens at a late layer on a patch of a church tower (first row), LATENTLENS provides highly accurate nearest neighbors across all layers such as "large tower with **clocks**". We also note that for LATENTLENS, the magnitude of the cosine similarity typically increases at deeper layers, i.e., the visual token representations become more similar to their textual nearest neighbors. For LogitLens, on the other hand, a higher similarity score (logit) does not necessarily translate into more interpretable descriptions. Furthermore, LogitLens descriptions often include unrenderable tokens, subwords, punctuation, and unrelated non-English tokens in Chinese. With LATENTLENS, however, we can simply merge the top matching token into a full word with adjacent subwords (since we have access to the entire sentence), if necessary, such as "b + **elf** + ry" into "belfry". Overall, these examples highlight the advantage of LATENTLENS: full-word and even sentence-level descriptions are more interpretable than subword tokens. We quantify this full-sentence advantage in Section E.11.

Visually rendered text. Images containing rendered text yield consistently

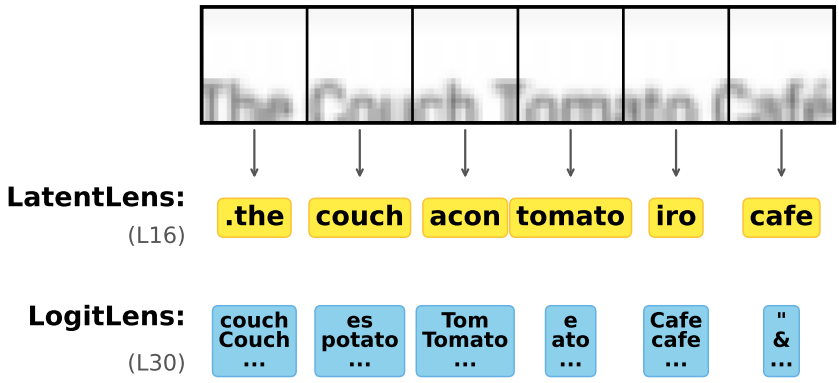


Figure 5.7: LATENTLENS vs. LogitLens results on rendered text for OLMo + CLIP ViT-L/14-336. LogitLens predicts plausible next tokens. We omit the surrounding sentence-level context for simplicity of visualization.

interpretable results when using LATENTLENS. In Figure 5.7, we show an exam-

ple for OLMo-7B + CLIP ViT-L/14-336. With LATENTLENS, the highest scoring contextualized representations correspond exactly to the words in the image.[10] When using LogitLens, on the other hand, we observe plausible next-token predictions which do not necessarily indicate what the visual token *intrinsically* encodes. For example, on the visual token where LATENTLENS predicts **couch**, LogitLens would predict **es** or **potato** instead. In other instances, LogitLens would even apply "correct" next-token-prediction on the rendered text and predict **Tomato**.

5.6 Related Work

We situate our work in the literature on understanding the mapping between vision and language models, the interpretability of VLMs, and analysing LLM representations.

Connecting (frozen) vision and language models. Several works have shown that LLMs can easily be adapted to process multimodal inputs (Tsimpoukelli et al., 2021; Mañas et al., 2023; Merullo et al., 2023; Lu et al., 2022), e.g., via small MLP or attention-based modules. Current state-of-the art VLMs (Yang et al., 2025; Deitke et al., 2025; Li et al., 2025a) follow the same mapping paradigm but include more sophisticated image token pre-processing, adaptation training stages or connector designs. Some works, in spirit related to us, have also explored representing visual tokens explicitly as a weighted sum of the LLM vocabulary (Liao et al., 2025; Masry et al., 2025) or as discrete tokens (Chan et al., 2024), instead of a vanilla MLP projector. Various works study why an LLM can so easily adapt to non-linguistic inputs: Lu et al. (2022) frame LLMs as "universal computation engines" that can process any data sequence with minimal weight updates. Patel and Pavlick (2022) argue how LLMs, only trained on language, nonetheless learn implicit world models of the physical world, e.g., of color RGB spaces. Han et al. (2025) disentangle the visual priors learned during LLM's text-only pre-training into perception and reasoning priors. Our setup

[10] For DINOv2, we would see generic words, e.g., "screenshot".

of understanding the frozen alignment between vision and language models is most similar to (Merullo et al., 2023), (Lu et al., 2022) and (Shukor and Cord, 2024), all of which explicitly keep the LLM frozen to characterize how vision integration is nonetheless possible.

Interpreting VLMs. Various interpretability methods, often initially developed for unimodal models (nostalgebraist, 2020; Cunningham et al., 2023; Belinkov, 2022), have been applied to VLMs (Neo et al., 2025), e.g., to characterize cross-modal concepts (Papadimitriou et al., 2025), cross-modality circuits (Nikankin et al., 2025), and the role of attention mechanisms in extracting information from visual tokens (Neo et al., 2025; Zhang et al., 2025). Most closely related to us are works that ask what a given visual token encodes in VLMs. Aside from training-based approaches like probes on visual tokens (Fu et al., 2025), many approaches leverage the inherent embedding space of the LLM to interpret visual tokens: E.g., Mokady et al. (2021) and Jiang et al. (2025a) study whether the LLM embedding matrix can interpret visual tokens, but only show qualitative examples on 1-2 models. LogitLens has been more widely used to interpret visual tokens at later LLM layers and applied to reduce hallucinations (Neo et al., 2025; Jiang et al., 2025b; Shukor and Cord, 2024; Park and Li, 2025; Wu et al., 2025). However, these works often involve either only small studies to quantify the interpretability via LogitLens, or only few models and with a closed set of object classes as labels. Overall, neither of these works compare the interpretability via the embedding and unembedding matrix (Logitlens). To the best of our knowledge, no prior work has considered contextual embeddings for this purpose. Notably, Phukan et al. (2025) leverage the *average* intermediate contextual embedding of the generated answer to mitigate hallucination on VQA. They do not investigate the interpretability of visual tokens, and only rely on contextual embeddings from a single generated output (not a large collection). Past work also leverages a pool of text descriptions to interpret image tokens or contributions of model components in CLIP (Chen et al., 2024; Gandelsman et al., 2024).

Another fundamental question is how vision and language embedding spaces relate to each other, such as characterizing a fundamental "modality gap" (Liang et al., 2022; Jiang et al., 2024a) or narrow-cone effects (Shukor and Cord, 2024) in models. Our findings can also broadly be seen as further evidence for a high structural similarity of vision and language representation spaces (Li et al., 2024b), coined as the Platonic Representation Hypothesis (Huh et al., 2024).

Representations in LLMs. Prior work has investigated how other non-discrete tokens (i.e., soft prompts) or intermediate states are represented in LLMs. Soft prompts have been found to sometimes showcase interpretable neighbors, albeit generic concepts only broadly related to the task (Lester et al., 2021). Bailey et al. (2023) challenge this view of soft prompts as "word-like" units. Hidden representations and contextual word embeddings in language models have been extensively studied along axes such as layer evolution (Voita et al., 2019; Aken et al., 2020), embedding geometry (Ethayarajh, 2019) and semantics such as word sense disambiguation (Peters et al., 2018; Eyal et al., 2022; Wiedemann et al., 2019; Chang and Chen, 2019). Notably, Eyal et al. (2022) also leverage a large index of contextual word embeddings linked to the sentence they occurred in.

5.7 Discussion and Conclusion

In this work, we established that visual tokens are consistently interpretable across LLM layers, even at the input to the LLM. The ability to find interpretable visual tokens relies on using contextualized textual token representations, instead of the input or output embedding layers of the underlying model. Our findings challenge the inconclusive results of previous work in a systematic study across interpretability methods, models, and layers.

These results also help explain why connecting separately trained vision encoders and LLMs requires only minimal weight and architecture updates. For example, one piece in the puzzle for explaining this straightforward mapping is our *Mid-Layer Leap* observation: visual tokens at the input are already aligned

with semantic intermediate language representations (e.g., layers 8–16), rather than word-level embeddings.

Broadly, it is a long-standing goal to understand the relation between the physical world and abstract symbolic processing, both in human cognition (Harnad, 1990) as well as AI systems (Huh et al., 2024; Bisk et al., 2020a). Our empirical study contributes toward understanding how visual and linguistic representations interface in neural systems, and to what extent we can find isomorphisms.

We foresee many exciting avenues for future work. A fundamental question is to what extent vision and language representations share deeper structural similarities beyond interpretability. LATENTLENS may extend beyond VLMs to other non-linguistic tokens such as soft prompts (Lester et al., 2021), latent thinking (Hao et al., 2025), or speech (Ògúnrèmí et al., 2025). It could also be applied to natively multimodal models (Team, 2024; Zhou et al., 2025; Team et al., 2023) or refined as a tool with a dynamic corpus.[11] In terms of downstream implications, we see promise in mitigating hallucination (Jiang et al., 2025b), as well as causal ablations investigating how interpretable vs. non-interpretable visual tokens affect task performance.

Limitations

We acknowledge several limitations of our approach. First, LATENTLENS requires pre-computing and storing a large corpus of contextual embeddings, which incurs storage overhead compared to methods like LogitLens that rely solely on model weights. Second, our interpretability judgments may be influenced by the Visual Genome corpus of sentences used for contextual embeddings. Third, we crucially have not studied how or if certain LATENTLENS interpretations affect the downstream behavior of the VLM. Finally, while we study 15 VLM configurations, our findings may not generalize to architectures that differ substantially from the transformer-based models we analyze.

[11] We present an initial exploration of this in Section E.12

Acknowledgments

We would like to thank our many colleagues from McGill NLP and the wider Mila community for their valuable feedback and brainstorming. BK is supported by the Vanier Canada Graduate Scholarships. DE was funded by IVADO Thematic Semester on Autonomous Agents and by research grant (VIL53122) from VILLUM FONDEN. MM is supported by the Mila P2v5 grant and the Mila-Samsung grant. SR is supported by Canada CIFAR AI Chairs program and IVADO R3AI. Finally, we thank Sonia Joseph and Elinor Poole-Dayan for final feedback on the draft.

Part III

Discussion and Conclusions

6

Discussion

This chapter first summarizes the contributions of each paper from the perspective of 2026: In retrospect, what were the key contributions of each paper and what aspects stood the test of time? How was the published work further used and expanded on, either by other researchers or myself? Then, in Section 6.6, I will envision future directions for the field of vision-and-language focusing on the theme of unification as well as interpretability. I will also reflect on what fundamental challenges still remain for the field, despite regular model releases with increasingly general and multimodal capabilities.

6.1 The Difficulty of ImageCoDe

Recap

The ImageCoDe task introduced in Chapter 2 is built around a two-player crowdsourcing game in which one annotator describes a target image such that a

second can identify it from ten candidates drawn from nearby video frames. This design yields visual minimal pairs with a uniquely strict definition of minimalness, and naturally elicits a diverse range of vision-and-language phenomena without explicitly prompting annotators. All models fell far short of human performance: zero-shot CLIP reaches 22.4% and fine-tuned models stay below 30%, against a human ceiling of 90.8%.

Key Contributions in Retrospect

IMAGECODE is unique among vision-and-language datasets in its operationalization of visual minimal pairs: by sourcing candidates from video frames rather than curated static images, it achieves a stricter definition of minimalness than most prior or concurrent work (Thrush et al., 2022b; Wang et al., 2023b; Awal et al., 2024; Suhr et al., 2019). Moreover, most popular vision-and-language datasets at the time provided generic, and often short, captions of an image. ImageCoDe's captions are much longer (often several sentences) and are highly context-sensitive: capturing the pragmatics of the two-player game and the diverse visual and temporal context of the nine distractor images.

One contribution that has aged particularly well is the long-term difficulty of the benchmark, thus continuously measuring progress of vision-and-language models. In AI, benchmark saturation within a year of publication is a common issue, yet even recent frontier models that I tested while writing this thesis fall short of human performance on ImageCoDe: On the original ten-image task, the hardest setting, the latest frontier models remain far from humans: on a random sample of 150 validation examples,[1] Gemini 3.6 Flash and Gemini 3.1 Pro reach only 68% and 61% accuracy overall, against a human ceiling of 90.8%. Consistent with the original paper, this difficulty is concentrated in video frames: accuracy on the video subset falls to 63% and 56% respectively, whereas static-image examples are nearly solved (100% and 95%). Notably, the larger Gemini 3.1 Pro

[1] I evaluate on a random sample of 150 validation examples, presenting all ten candidate images together and asking the model to select the target frame; setup and code at `https://github.com/mcgill-nlp/imagecode`.

does not surpass the smaller Flash model on this task. We note that alternative, more model-friendly formats are possible: presenting all ten images at once may overload the model's context, and requiring it to output a frame number may not be the most natural response format. Nonetheless, this setup reflects the most widespread way these models are used today, through their chat interface, and therefore represents a typical interaction a user would have on a task like IMAGECODE with frontier models.

What is now outdated. With papers as "old" as ImageCoDe in our fast-moving field, it is also fruitful to reflect on which aspects are now considered outdated. The community standard is now to evaluate in out-of-distribution zero-shot settings, so releasing a train-validation-test split is no longer the norm.

Impact of ImageCoDe

Since IMAGECODE's release in 2022, several works have built on top of it: Ou et al. (2023) generate RSA-based pragmatic captions on our data; Gaur et al. (2025) reach the current captioning state-of-the-art of 37.9%; Dessì et al. (2022) show that neural captioners develop non-human-interpretable communication patterns that a CLIP-listener can exploit; and the training split has been incorporated into LLaVA-OneVision (Li et al., 2024a) and Mantis (Jiang et al., 2024b). Related contributions around minimal pairs (Awal et al., 2024) and multi-image reasoning (Ross et al., 2026) have since followed.

Where did the field move from here? With the impressive progress of LLMs, we can ask which of the two modalities is holding back vision-and-language understanding. The language aspect is closer to "solved": text-only compositionality, long coherent generation, many-turn dialogue, pragmatic and intent understanding (OpenAI, 2024; Akhtar et al., 2026; Ruis et al., 2023). Hence, we see a shift towards benchmarks that test pure *perception* (without any complex language component) in multi-modal LLMs, such as Blink (Fu et al., 2024) or IntPhys 2 (Bordes et al., 2025). Despite early awareness of models relying on text-only shortcuts such as on the original VQA task (Goyal et al., 2017b), studies

regularly expose the very same issues in subsequent benchmarks and evaluation protocols (Hsieh et al., 2024; Asadi et al., 2026; Krojer et al., 2025).

After static image tasks posed the main challenge for many years, videos are emerging as the next major challenge, closing the gap to the kind of perceptual and temporal data children are exposed to: moving scenes that capture richer semantics of the physical world (Zellers et al., 2022). While video models at the time of ImageCoDe were tested on simpler videos and actions (Xu et al., 2016; Ji et al., 2020), often in practice resorting to single-frame biases (Lei et al., 2023), we have seen recent progress on harder and more genuine tests of video understanding such as intuitive physics (Assran et al., 2025). It is nowadays common for strong multimodal models to support multi-image input (Yang et al., 2025; Team et al., 2024) but ImageCoDe's set of 10 candidate images still constitutes a relatively large and challenging visual context. Finally, after the release of the ImageCoDe task, a related line of work picked up momentum: visual search guided by an LLM reasoning process (Wu and Xie, 2023). On the other hand, fields such as visually-grounded pragmatics and emergent communication did not gain much traction after early interest during the pre-LLM era (Andreas and Klein, 2016; Lazaridou et al., 2018).

6.2 Progress in Text-Guided Image Editing

Recap

In Chapter 3 we presented AURORA as a combination of curated training data, a capable image editing model and a benchmark. We fixed two aspects of the data on which image editing models are trained: First, the data focuses on simpler object and attribute edits, so models performed poorly on temporal actions and physical reasoning. Second, we identified noise issues in existing training examples (image pairs and a prompt) where too many changes are depicted in the target image compared to what the prompt demands. Our final model is trained on more than 280K minimal-change examples, spanning actions and reasoning, and exceeds existing models at the time on more challenging

temporal and spatial edits.

Key Contributions in Retrospect

At the time of publication, AURORA addressed a real issue with image editing models: general text-guided image editing models were largely limited to object- and attribute-level changes, because action- and reasoning-centric training data was hard to curate well. Together with concurrent work (Souček et al., 2024; Black et al., 2024), we added evidence that videos (even naive selection of first and last frame of the video) can provide rich supervision for image editing. The primary contribution is data-centric: by carefully curating minimal pairs from videos and simulation engines, we demonstrated that such edits can be learned, and that the data bottleneck was a major initial obstacle rather than the architecture. Our evaluation insights around hackable metrics have aged well: judging fine-grained semantic details of generated or edited images remains non-trivial (Ku et al., 2024).

What is now outdated. As a model, AURORA's performance on action-centric editing is now far behind recent image editing models (Deng et al., 2025) and closed-source general-purpose ones (Team et al., 2024). However, predicting how a scene changes based on actions and physical dynamics is still a challenge, compared to stylistic or object identity changes. We illustrate this in Figure 6.1 with Gemini 2.0 Flash on an AURORA-Bench example where models in our study failed: While the model is now moving the correct hand in the correct direction without any visual artifacts, it still fails to put it exactly where one would turn on a faucet.

How Was AURORA Used? Where Did the Field Go?

Edits that require an understanding of real-world dynamics (actions, physics, causality) remain hard, and AURORA paved the road for further progress in this direction, while also being used as training data for unified generative models (Tong et al., 2025; Ahmadi et al., 2026). The idea of using video frames as a source of training data was explored in concurrent as well as subsequent work

Instruction: "Turn on the water tap with the left hand"

Figure 6.1: An AURORA-Bench example (EPIC-Kitchens subset) edited by Gemini 2.0 Flash.

(Souček et al., 2024; Black et al., 2024; Mahajan et al., 2025). As video generation models become more capable, recent work explicitly generates the whole video as intermediate reasoning steps to arrive at the final edited image (Wiedemer et al., 2025; Rotstein et al., 2025), instead of implicitly leveraging video frames only as training signal. Moreover, the focus of image editing community has shifted from simple edits to various forms of reasoning such as reasoning about abstract knowledge or logic (Deng et al., 2025; Zhao et al., 2026).

More broadly, action- and reasoning-centric image editing connects to a deeper problem: world modeling, i.e., predicting the next observation after an action is taken. AURORA's form of image editing can be seen as one-step controllable video generation, which in turn can serve as a generative world simulator (Yang et al., 2024b; Xiang et al., 2024; Zhou et al., 2024). It is a highly debated question to what extent learning to generate next frames (video generation) leads to a consistent physical world model (Yang et al., 2024b; Kang et al., 2025; Bansal et al., 2025).

6.3 Unifying Visual Understanding and Generation

Recap

With DiffusionITM in Chapter 4, we empirically showed that an image generation model can implicitly also act as a discriminator, and thus a vision-and-language understanding model. Concretely, we interpret the noise prediction objective of Stable Diffusion (Rombach et al., 2022) as a score for how well an image and text semantically match (ITM). Our method not only enabled text retrieval settings but also image retrieval, and we further improve DiffusionITM with a novel hard-negative finetuning approach. Across 8 diverse benchmarks for vision-and-language understanding, we show that an image generation model can be on par with CLIP (Radford et al., 2021a). Our study thus enables a direct comparison between generative and discriminative modeling paradigms on the same vision-and-language benchmarks.

Key Contributions in Retrospect

Our study was motivated by public excitement around the seeming compositionality of DALLE-2 with its famous "avocado chair" example (Heaven, 2021; Hill, 2022). However since it was non-trivial to systematically evaluate such a capability, an approach like DiffusionITM was needed to ground the debate in experiments: A model like Stable Diffusion can perform similarly to existing models on a compositionality benchmark like Winoground (Thrush et al., 2022a) but not significantly better. For a brief period, very similar research questions as ours received excitement from the community: How can we leverage the impressive capabilities of Stable Diffusion and Co. for other tasks in computer vision and multimodality (Li et al., 2023a; Jaini et al., 2024; Burgert et al., 2022)? While this excitement faded, the broader question motivating DiffusionITM has remained central to the field: Visual understanding and generation have historically relied on separate encoder representations and architectures (e.g., CLIP for discrimination and as input for LLM captioning, VAEs for generation), but can we unify them? Two additional aspects of the DiffusionITM contribu-

tion had potential to have long-term impact but never gained momentum: a) finetuning diffusion models with more understanding-based objectives and b) automatically evaluating the understanding capabilities of generative models via discriminative tasks.

What is now outdated. Broad adoption of diffusion representations for understanding tasks has not materialized. Extracting useful representations from diffusion models is in practice finicky since design choices such as at which denoising timestep to extract features matter significantly. The field has moved instead toward explicit joint representation learning (Zheng et al., 2026) and architectures (Team, 2024) that unify understanding and generation from the ground up, rather than repurposing generation models post-hoc. To get the best of both worlds for visual understanding and generation, models are now often trained with diffusion and autoregression objectives in parallel (Zhou et al., 2025; Tong et al., 2026).

How Was It Used? Where Did the Field Go?

Post-publication, harnessing diffusion representations for downstream tasks remained an active thread: Qu et al. (2024) directly built on our discriminative finetuning of Stable Diffusion and used it to improve generation; Burgert et al. (2022) repurposed Stable Diffusion as a segmentation model; and Jaini et al. (2024) found that generative classifiers exhibit human-like shape bias, in contrast to their discriminative counterparts.

Yet as the original paper captures, the Generative AI Paradox (West et al., 2024) applies here: our results show that diffusion models converted to discriminators do not consistently outperform CLIP on compositional ITM tasks, and newer generation models (e.g., SD-XL (Podell et al., 2024)) sometimes perform worse despite better image quality. Being able to generate does not straightforwardly imply being able to discriminate or reason, which has held up as a recurring theme in the field.

6.4 Interpreting Multimodal Representations in LLMs

Chapter 5 contains my most recent publication (Krojer et al., 2026), accepted at ICML 2026 just before the completion of the thesis. So it is yet to be seen how the tool we proposed (LatentLens) will be used and where the field will move. Thus instead of following the structure of the above three sections, I will solely reflect on the key contributions and speculate on potential impacts.

Key Contributions

The central contribution of LATENTLENS is a training-free interpretability method for visual tokens in multimodal LLMs. Rather than comparing visual token representations against the LLM's static embedding or unembedding matrices (as with established EmbeddingLens and LogitLens), LATENTLENS retrieves nearest neighbors from a *contextual* representation space constructed from tokens at the LLM's intermediate layers. This change of comparison space substantially increases how many interpretable visual tokens we can find: while prior methods label only 23–30% of visual tokens as interpretable, LATENTLENS reveals that the majority (72% on average) are interpretable, consistently across models and layers. LogitLens is one of the most widely used tools in (mechanistic) interpretability (nostalgebraist, 2020) and has specifically been applied to probe visual representations (Jiang et al., 2025b; Neo et al., 2025; Kim et al., 2026). Showing that such a widely used tool can substantially underestimate the interpretability of tokens is a key insight for practitioners and researchers in the "science of LLMs". A second notable contribution is the breadth of evaluation across model configurations. Many interpretability studies draw conclusions from one to three models, which limits the generalizability of findings. Our study instead evaluates across a 3×3 grid of vision encoder and LLM combinations, runs extensive training ablations, and probes six off-the-shelf VLMs. The stark differences we observe across these configurations, especially with our baselines EmbeddingLens and LogitLens, underscore that interpretability findings can be highly model-specific, and that results from a narrow set of models do not

reliably generalize. On other hand, our breadth of evaluation settings now allows us to be confident that LatentLens, in contrast to our baselines, works in most of the settings.

Future impact and applications.

So far, Chapter 5 provides concrete evidence that LatentLens outperforms tools like Logit Lens (nostalgebraist, 2020) for image inputs to LLMs. An exciting future question is whether LatentLens will outperform or complement Logit Lens in more scenarios beyond images, either on text tokens themselves or on other types of non-textual tokens. Two recent pre-prints have already successfully adopted LatentLens, either for interpreting compression tokens for extracting embeddings from LLMs (Adlakha et al., 2026) or gender bias in VLMs (Marin-Llobet et al., 2026). My ongoing follow-up work is further investigating what insights LatentLens can provide across various other settings where LLMs process non-discrete tokens: audio (Chu et al., 2024), videos (Clark et al., 2026), latent thinking tokens (instead of verbalized Chain-of-Thought) (Zhang et al., 2026), VLA action tokens (Kim et al., 2024), soft prompting (Lester et al., 2021) or spatial pointer tokens (Su et al., 2026).

Finally, interpretability has grappled for many years with questions of faithfulness (Saphra and Wiegreffe, 2024; Wiegreffe and Pinter, 2019): If a study claims that a model is solving a task or representing a concept in a certain way, is this merely a by-product or causally and systematically affecting final model outputs? Our LatentLens method is promising and similar in principle to Logit Lens, which has been applied successfully to steer model behavior (Jiang et al., 2025b). However, future work still needs to convincingly prove the faithfulness of LatentLens.

6.5 Takeaways and Limitations Across the Thesis

The main theme of this thesis is the deep integration of vision and language in AI models. However, the term *integration* is deliberately kept broad, and fully characterizing it involves many facets. Various aspects of the ImageCoDe task

in Chapter 2 arguably require deep modality integration, if one considers how a human might solve such problems: Our *representations* of the visual input are highly dependent on the task and language input, i.e., depending on the contextual description and what we know about the guessing game setup, we would focus on different images and different parts of each image. In contrast, many models at the time such as CLIP (Radford et al., 2021a) or today's models (Liu et al., 2023; Wang et al., 2024b) encode these images statically. A human would also integrate the modalities during their *reasoning*, starting with an initial guess based on the description, zooming into parts of the images, going back and forth between description and vision, and so on. Thus, ImageCoDe was a good starting point to quantify various aspects of how a model with deep integration should behave. However, Chapter 4 showed that even newer generative models still struggle on many tasks similar to ImageCoDe. Here, similar to Chapter 3 on image editing, we study deep integration by unifying traditionally separate paradigms in vision-and-language: visual understanding (often more language-centric) and visual generation (often more vision-centric). Finally, Chapter 5 approaches the deep-integration question head-on: Are the models internally representing vision and language inputs in some kind of shared representation space? With LatentLens, we find that visual tokens, processed by LLMs, are more interpretable as language than previously thought, with even strong alignment in the earliest LLM layers.

Beyond the primary question of modality integration, we can identify several secondary takeaways: Across three of the four papers (Chapters 2 to 4), the most fruitful idea is isolating one meaningful variable at a time via minimal pairs: It allows testing genuine skills more reliably (Chapters 2 to 4) and can serve as a clean supervision signal (Chapters 3 and 4). Curating such minimal pairs, especially when minimally changing visual aspects, is non-trivial and can involve extensive crowdsourcing efforts, e.g., when collecting the ImageCoDe and AURORA datasets. In both cases, video frames were rich sources of meaningful minimal changes.

Limitations. Considering the fast pace of our field since the beginning of this thesis in 2021, one should also reflect on broader limitations and outdated aspects of the thesis as a whole:

- **A more systematic treatment across paradigms.** The question of modality integration could have been tackled more uniformly across architectures: for example, applying a LatentLens-style analysis not only to multimodal LLMs but also to natively unified models such as Chameleon (Team, 2024) or Emu3 (Wang et al., 2024c), as partially explored by Serra et al. (2025).

- **A more behavioral formalization.** In ImageCoDe, integration could have been pinned down more precisely at the behavioral level: which sub-skills genuinely require deep integration, and which could be solved by shallower models?

- **Reasoning- and tool-augmented models.** Today ImageCoDe would be studied with chain-of-thought models and agents capable of zooming or tool use, rather than the single-pass models of its time.

- **The shift from IID to OOD evaluation.** Training and testing ImageCoDe and AURORA on their own in-distribution splits is now outdated, as the community has largely moved to out-of-distribution evaluation, with the caveat that current models may have seen many academic benchmarks during pre- and post-training.

- **Scale.** None of the papers focus on scalability: do larger models fare better on these tasks, does collecting ever more video frames keep improving the AURORA editing model, and how does the interpretability measured by LatentLens change with larger or newer models? We only slightly touch on the last question in Chapter 5, finding some evidence of reduced interpretability in larger models, though not conclusively.

6.6 A Roadmap for the Field

When motivating this thesis in Chapter 1, I observed that the field is far from converging on the best way to unify vision and language in a single model (illustrated in Figure 1.1). Compared to LLMs, where the field is seemingly more settled on its modeling paradigm, here we are faced instead with a myriad of scientific questions and practical choices. With this in mind, I will lay out a roadmap for the future of multimodal models, with emphasis on how to improve and measure integration of modalities (exemplified by recent unified modeling efforts (Team, 2024; Tong et al., 2026)), how we can scientifically study model internals as well as what interpretability toolkit is needed to study these multimodal questions.

Are we on the right path for unified models? This is the foundation on which the other parts of the roadmap rest. We need more systematic studies ablating the various aspects of unified modeling to understand the trade-offs: 1) Should we train from scratch on multimodal data or benefit from already strong unimodal backbones that we can post-hoc align? 2) How do we combine autoregression and diffusion in a single sequence of multimodal tokens? 3) To what extent should we share weights across modalities or instead reserve specialized weights for each one? Recent principled studies that explore this design space and multimodal scaling laws (Tong et al., 2026; Shukor et al., 2025) are a step in the right direction, albeit computationally nearly impossible on a purely academic budget. With recent releases of "partially" unified models (Deng et al., 2025; Liang et al., 2025), it remains an open question whether full unification is simply infeasible or whether we merely lack key insights about how vision and language representations interact. The ultimate check for whether unified models are *actually unified* will be whether the learned representations and mechanisms (e.g., *circuit discovery* from mechanistic interpretability (Conmy et al., 2023)) are unified according to certain metrics of embedding spaces or overlap (Serra et al., 2025; Nikankin et al., 2025). These metrics would measure,

on a high level, whether concepts and computations are implemented cross-modally or some modality-agnostic way rather than in separate subspaces. Tools in the spirit of LatentLens (Krojer et al., 2026) will be essential for measuring this.

Downstream of more unified models. Once we have models with more unified representations and internal machinery, two concrete downstream capabilities become more tractable. First, unified representations should enable better multimodal embedding extraction. Here, the concrete challenge is to summarize a large context (e.g. a document) of interleaved text and vision tokens into a single vector, as required for retrieval tasks (Jiang et al., 2025c). Second, multimodal chain-of-thought reasoning (Bigverdi et al., 2025; Li et al., 2025b) will become more seamless when such reasoning can be conducted over an underlying representation space that is more unified and less text-centric. Imagine a model that can flexibly choose the most suitable token type at each reasoning step (text, visual, latent, audio tokens) rather than being constrained to purely textual reasoning traces (Su et al., 2025). This would be particularly helpful for tasks such as ImageCoDe (Krojer et al., 2022) from Chapter 2, where the model has to go back and forth between visual context and language, such as zooming in and considering and comparing different counterfactual images.

Synergy. In contrast to the more internal, representational signs of unification discussed above, perhaps the most compelling validation would come from strong *behavioral* evidence of synergy in both directions: Joint training on vision and language should benefit each modality more than training on either alone. It is well-established that language helps vision such as CLIP's improved image classification due to language supervision (Radford et al., 2021a), or more recent open-ended image segmentation (Kirillov et al., 2023). However only sparse evidence exists for the other direction of synergy, perceptual grounding helping language understanding (Zhuang et al., 2024; Qin et al., 2026; Tong et al., 2026). Based on intuitions from human learning where grounding in a physical environment is critical for language (Bisk et al., 2020a), we would expect that

joint training with vision would also lead to better language performance, for example on physical commonsense (Bisk et al., 2020b). Monitoring the training dynamics of joint multimodal training would allow us to measure if and how such synergies emerge (Team, 2024).

The dynamics of the world. Truly understanding the physical world goes far beyond static images, and might require learning an internal world model, i.e., predicting the next physical state given an action on the previous state (Ha and Schmidhuber, 2018; Assran et al., 2025). Arguably, many physical concepts may not even be easily expressed in language: Captioning a static image or answering questions with language-expressible concepts works well within the current multimodal LLM paradigm, but it is less clear that this equally applies to reasoning about intuitive physics or modeling the continuous dynamics of a scene (Garrido et al., 2025; Krojer et al., 2025). Understanding how intuitive physics and world models are implemented in the weights of video models is therefore a next major scientific question (Joseph et al., 2026), and language may play a smaller role here than it does for static image understanding. This in turn suggests that we may need very different interpretability approaches to study world modeling, outside the LLM- and safety-driven mainstream approaches of mechanistic interpretability (Elhage et al., 2021; Rai et al., 2024). This direction connects back to work presented in this thesis, where we found that videos provide a rich learning signal and unique reasoning challenges (Chapters 2 and 3).

Interpretability tools tailored for multimodality. All of the above directions require a reliable interpretability and analysis toolkit that can facilitate scientific insights across many modalities and modeling paradigms. LatentLens in Chapter 5 or DiffusionLens (Toker et al., 2024) for text-to-image models were steps in this direction: the dominant interpretability tool developed with text-only LLMs in mind, such as Logit Lens (nostalgebraist, 2020), needed to be revised for non-LLM contexts. As the field moves towards models and paradigms that are natively multimodal, many more steps towards modality-general interpretability

tools are needed. While this is challenging, it is also exciting for the field of interpretability. We can now pose questions that were not conceivable in uni-modal models: Which parts (e.g. circuits) of the model specialize on modalities, and which are modality-agnostic (Nikankin et al., 2025)? How do models handle and prioritize conflicting evidence from different modalities (Chen et al., 2026)? And which concepts will be represented in their unique modality space vs. in a shared abstract space?

7

Conclusion

Across four papers, this thesis approached the deep integration of vision and language from three angles: evaluation, modeling, and interpretability. Over the span of this thesis (between 2021 and 2026), multimodal capabilities and the focus of the field have progressed immensely: We can now converse about minute visual details of any image with a chatbot and, in the same interaction, also ask it to generate novel images, usually involving some form of compositionality or reasoning (Team et al., 2024). In Chapters 2 and 3 our evaluations highlighted the challenges models at the time faced when reasoning over vision and language together. Yet even to this day, many fundamental failure modes and shortcuts persist (Asadi et al., 2026; Fu et al., 2025). In Chapter 4 we developed DiffusionITM, which enables an image generator to also act as an image understanding model. While this particular paradigm did not mature, the quest to "unify everything" in one model is still far from solved (Tong et al., 2026). Finally, in Chapter 5 our interpretability method (LatentLens) revealed how

visual tokens are interpretable as language in the LLM embedding space. Yet, we still know very little about the exact internal mechanisms of unifying vision and language, and the field of interpretability is still debating the very foundations of what it studies: what does it mean for a token to be "interpretable", for a circuit to be meaningful, or for a model to "represent" a concept at all (Lipton, 2018; Sharkey et al., 2025)?

Stepping back, how vision and language relate inside a model raises questions that go far beyond engineering: What *are* perception and language (for humans and for AIs), beyond their clearly different surface form of either pixels or words? Could reasoning eventually take place in a fully cross-modal latent space? And how might all this ultimately bring us closer to a truly embodied, grounded understanding of the world? This thesis is one small contribution to this broader conversation that spans across scientific and philosophical disciplines.

Part IV

Appendix

A

Behind the Scenes

Around the middle of my PhD, I decided to include a *Behind the Scenes* section in the Appendix of the AURORA paper (Chapter 3). Since then, including such a section has become a tradition for all my first-author papers, and I am happy to see collaborators pick up the habit as well. These sections in some way or another tell the story behind the paper, the ups and downs, the initial brainstorming and the real motivation of the authors. The hope is that this humanizes the field and helps young researchers see that behind the polished work is a lot of confusion, uncertainty and us humans. I encourage the reader to have a look at these Behind the Scenes sections in Sections C.13 and E.15, as well as in papers not included in this thesis (Krojer et al., 2025; Krishnan et al., 2025; Ahmadi et al., 2026; Vergara-Browne et al., 2026; Tür et al., 2026).

This is a special Behind the Scenes section as it spans so much more than a single paper. While I will talk less about the scientific details of each paper, I will follow a similar structure (just one abstraction level higher): 1) What drove me

to start this project (i.e., the project of spending five years on a PhD) and what were the initial expectations?; 2) What did I end up working on, with whom did I work and how did my ideas develop over the years?; 3) Finally, general lessons learned and what comes next academically.

A.1 Applying for PhDs

As an undergrad I found my PhD direction because I was unhappy with the state of the NLP field in 2020. But let's start at the beginning: After dabbling for one semester in a pure math major, I got my BSc in computational linguistics at LMU Munich. It was an easy workload, allowing me to work on the side at a computer vision startup, start a philosophy society at university and to get early research experience in NLP. My first research experiences during undergrad were quite coincidental: I did not know that Hinrich Schuetze (who supervised my bachelor thesis) was a "big name" in the field. I simply chose the local university at the time and a major that sounded multi-disciplinary. And nobody could anticipate that NLP would soon become the most trendy topic as the birthplace of LLMs[1]. I was not really blown away by the topics in NLP research yet but I think deep-down I knew that research in general would fit me very well; and I was naive and driven: I really wanted to get papers out and a chat with a visiting PhD student (Denis Peskoff) convinced me that it would be exciting to apply directly for a PhD in North America. That sounded much more exciting than doing an MSc in Europe. So I went to the two main professors in our department and asked them if they could supervise me with the goal of publication. They both matched me with some senior PhD students in their lab and somehow it all worked out: Two shared first-author papers got into smaller NLP venues. But as those papers were wrapping up in early summer 2020, I had not found a topic yet that felt worth it to spend five whole years on: Philosophically, just working on narrow NLP tasks such as translation did not feel like anything resembling the embodied intelligence and cognition we humans have. But if

[1] fantastic read on this: When ChatGPT Broke an Entire Field: An Oral History

I remember correctly, I was also too inexperienced in the literature to put this disillusionment into words. Instead I had to stumble upon the paper Experience Grounds Language (Bisk et al., 2020a) on Twitter to find my direction: Each word in this paper resonated so much with me! After less than a page, I kind of knew I had found something worth it to dedicate five years of my life to: Language grounding. It was philosophical (*Where do words get their meaning from?*), it was interdisciplinary (spanning NLP, computer vision, robotics, cognitive science) and it also felt timely when language models were taking off but it was unclear if they could reach "genuine understanding". Now I was in PhD-application mode, making spreadsheets that ranked different potential professors and locations. Many of the potential supervisors I reached out to and eventually applied to were from the long author list of (Bisk et al., 2020a) and almost all of them were working on language grounding or on some form of multimodal NLP. I attended my first conferences (virtual due to COVID). Twitter and cold emails played a big role, too. In the end I applied to around 15 places, all in North America except Edinburgh.

I share my full statement of purpose in Section A.5 for more details on my motivations and what specific projects I pitched to work on. Feel free to also read some of the blog posts I wrote at the time: What made me do a PhD far away? and Attending my first conference! ACL 2020 from an undergrad's perspective and ELI5: Language Grounding.

A.2 Five years at Mila

In the end I chose to do my PhD at Mila and McGill University in Montreal. There are too many factors going into such a decision to list them all here. A big factor was definitely that both Mila and the city of Montreal have a great vibe. My visit days were all virtual due to COVID and I would often instead watch YouTube videos of the respective cities to imagine a potential life there. My PI Siva also made it easy to commit to Mila and put effort into recruiting me.

It is interesting how almost nobody ends up working on what they envi-

sioned in their Statement of Purpose. I had big ambitions of working explicitly on the Big Philosophical Questions around language grounding, in all its facets: embodiment, interaction with other agents or humans, video understanding, grounding in time, comparisons to language acquisition in humans. In my Statement of Purpose, I expressed interest to work on "modeling how humans acquire language", emergent communication (Lazaridou et al., 2018), temporal commonsense (i.e., grounding in the passing of time), video understanding, developing benchmarks for grounded tasks like embodied navigation or instruction following, and even developing simulation environments for multi-player games to study the social interaction aspect of grounding. Of course, I ended up mostly working on other things.

The PhD started four months "early" with an internship in Siva's lab and naturally my first internship project was smaller-scoped and less philosophically ambitious, yet still very fun: Siva had an interest in minimally different pairs of images, either as a hard task or as a training signal for compositionality. I eventually proposed videos to find minimally different pairs and the human annotation protocol; that's how (Krojer et al., 2022) came about. In retrospect I would have done things very differently, starting with the title: the one we chose does not really tell you any of the unique contributions of ImageCoDe! Even a boring highly descriptive title like "Gamified Describing of Minimal Visual Differences from Nearby Video Frames" would have been better. I also dreamed of having even more than just sets of 10 minimally different images, why not more. More is always better right? In retrospect, choosing less than 10 would have been better probably. But the paper was a success and slowly caught some people's attention. It also helped that Siva insists that his students put in effort beyond just submitting the paper to a conference. In our case that meant setting up a leaderboard for the task or making a nice demo (on which I spent more than a week, in pre-Claude days). Since this was primarily a dataset contribution, it felt straightforward and gave me early confidence as I was just starting the PhD. While one should take risks in a PhD, I would recommend to try to have smaller

successes early on. Later PhD projects can be more open-ended and confusing. This is especially true when the student comes directly out of undergrad and I think this was also part of Siva's thinking.

In the first months of the PhD I was still brainstorming big ideas around grounding and would even have liked to work on some embodied systems. One of the craziest serious ideas I was sketching out at the time involved a longitudinal study over months where humans would be instructed to teach an embodied system language. They would take the little device through their daily life, point at things, use more complex language eventually, the system could actively ask for more information via pointing to get moved closer to something by the teaching human. This project would have involved all aspects of AI and I wanted to do it all: life-long learning, RL, robotics, video understanding, NLP, Human-Computer-Interaction. During that early excitement around grounding I also started the "Language Grounding Reading Group" that would run for three years at Mila and it would be the start of many deep discussions and friendships with, e.g., Oscar Manas and Rabiul Awal.

Then came the longest slump of my PhD, some form of mini-burnout and existential crisis. The crisis started with me questioning if I was doing any good for society by advancing AI models, and once I had made peace with some of that guilt, I questioned the whole thing: Am I doing a PhD for the right reasons (curiosity, helping science/society, fun), or because my self-worth is tied to being smart, impressive, high-achieving? It was important to work through these questions without flinching away from them but it also meant that my research progress was slower. Only much later, towards the last 1-2 years of the PhD, would I get back to the same level of excitement I had felt about research during those early PhD days.

So I did not end up working on the grounding question itself in my PhD and it is in many ways a very vague question, hard to operationalize and to make tangible progress on. A lot of the empirical assumptions were also wrong: LLMs alone could get us very very far on, e.g., commonsense or reasoning without any

perceptual or interaction-based grounding. As Jacob Andreas put it in a blog post on world models from 2024:

> We have vision-and-language models of the kind that grounding fundamentalists (I used to be one) said would fix the outstanding problems with text generation. But at scale, the addition of visual supervision doesn't seem to change the quality of text generation very much. So even with grounding—and in other domains like video generation or protein sequence modeling—we still can't agree whether neural sequence predictors are really understanding or shallowly manipulating their inputs.
>
> One of the biggest consequences of all this is that the community has started asking a better version of the Big Question: do LMs have internal world models?

I am glad I instead ended up choosing more well-defined problems that were feasible and in line with recent developments such as diffusion models or world models. After finalizing the whole ImageCoDe release in the spring of 2022, I unsuccessfully tried to make any follow-up ideas work for at least six months.[2] I was a bit adrift, unsure what I should work on next and also considered AI safety or ethics so that my work would be more closely aligned to positive societal impact. It was also the time when I took it slower to recharge and find my innate excitement again. On top, I took the time-intensive DL and RL classes in those semesters.

There was not one moment I could point to that got me out of the slump. It was a slow process of just sticking around, not overworking too fast again but allowing my curiosity to just wander, and having a wonderful social support network. A year after releasing ImageCoDe, I had to pick a class project for an

[2] For example: Can we make CLIP see more fine-grained differences by feeding it different crops instead of the whole image? Or, can we train a multi-task model on many multi-image datasets to improve upon the accuracy?

NLP course and both Siva and I had been excited by the progress of text-to-image models: Some of these outputs are so compositional! But how can we measure that or show that, e.g., Stable Diffusion might be more compositional than CLIP? The community knew that many vision-and-language models like CLIP would fail terribly on tasks like Winoground (Thrush et al., 2022a). This is how the second paper of my PhD, DiffusionITM (Chapter 4), happened. This was my most technically demanding work and I gained a lot of confidence from that. Attending my first NeurIPS (New Orleans) was great! DiffusionITM got me a bit closer to enjoying the process itself and the technical aspects again, and being grounded in contemporary questions of the literature — opposed to chasing an elusive big philosophical question. Similar to ImageCoDe, some questions from DiffusionITM felt unanswered and kept popping up in my mind. But again not much came out of it.

Now I had, in some ways, made good progress to graduate but, in some other ways, it felt like I was still playing catch-up to the level of excitement, dedication and clarity I had when starting the PhD. Chris Pal, a professor at Mila, had already been helpful on the DiffusionITM paper and stayed as a collaborator to brainstorm my next steps. Siva involved more senior researchers to supervise me such as Varun Jampani (at the time at Google). Looking back, I levelled up as a researcher in this period: How to think about picking a research problem, how to decide early when to move on from a project. Around that time, I started a Google Doc where I would add research ideas every few days, which is now, three years later, full with more than 40 idea pitches. I would develop a ranking system to judge ideas along several criteria:

1. **Excitement: are you excited deep down?**
2. Impact (Q: do we really think we can predict this as researchers?)
3. Risk of working out/uncertainty
4. Novelty/chance of being scooped

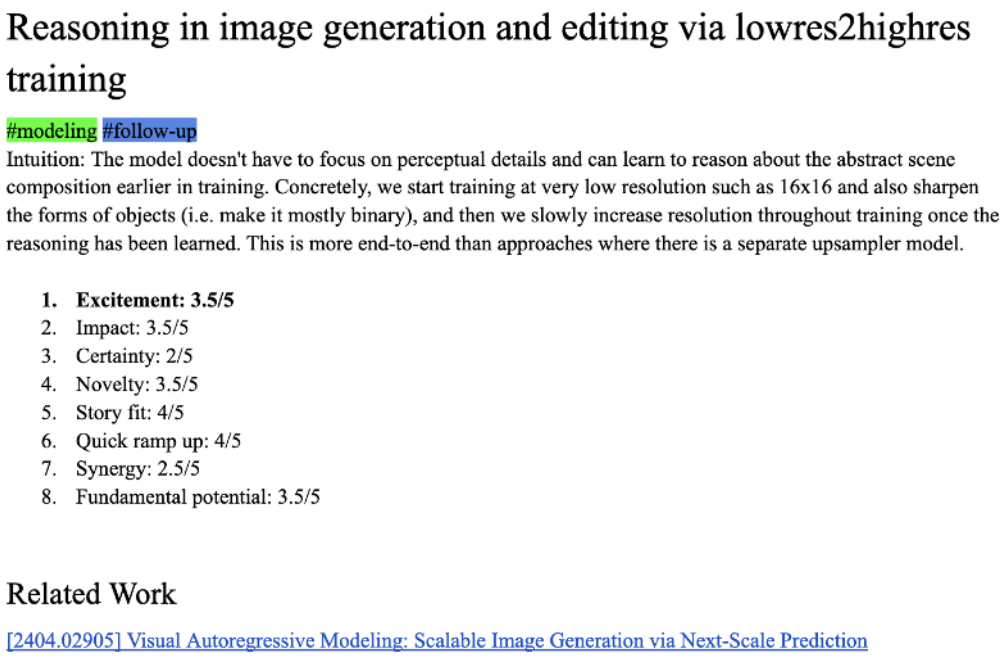


Figure A.1: Example of my paper rating system.

5. Alignment with current momentum of mine/direction + fits story of final PhD thesis
6. Method vs. Eval/Analysis ratio
7. Quick ramp up: Overhead to learn new topic/papers or set up environments
8. Synergy with Siva, lab members and collaborators
9. Fundamental potential: can I easily make this into a fundamental paper?

I would then assign a score from 1 to 5 for each criterion, as illustrated in Figure A.1. Going through such an elaborate process for writing down research ideas sounds like a lot but it only takes five minutes in practice and really makes you think about all aspects of the idea. It was a good exercise for me at this point

of the PhD and I would gradually do it less over the years, as I now internalize these questions implicitly. As I was brainstorming ideas and also implementing some aspects of them in winter 2023/2024, I discarded one idea after a month for not being impactful enough (analyzing part-whole relationship understanding in image generation models). Then, during one meeting with Chris Pal, we settled on the following simple idea: Can nearby video frames act as a rich supervision signal for image editing? Initially we were trying to show this with videos in the wild — either from movies or YouTube — but we soon realized how rarely videos in the wild depict a single meaningful change. So we pivoted to more curated videos from Action Genome and recruited crowdworkers on AMT. After ImageCoDe, and to a lesser extent even DiffusionITM, my project again involved extensive management of crowdsourcing. It takes a lot of iterations to get right, from making the task and instructions well-defined to communicating on a daily basis with the workers. I am quite proud of how I handled it for all the projects, with some crowdworkers becoming something like friends, even e-mailing me years later or expressing how much fun the work was. Anyways, once we had found the right videos and crowdworkers, we also added simulation data into the mix, because why not. This is how AURORA happened. With the main contributions being the data efforts, we submitted to the relatively new Dataset & Benchmarks Track at NeurIPS 2024. I was positively surprised how great the reviews were! First spotlight in my PhD.

I have mixed feelings about big-tech but I also wanted to experience one "proper" industry internship during my PhD. While my dream place was undergoing major changes and thus did not work out (AI2, winter of 2024), I ended up in a fantastic lab at FAIR (Meta). The JEPA team works primarily on world modeling and self-supervised learning (from images or videos), and there was lots of overlap with my supervisor Mido Assran. I regret not having thought more deeply about modeling projects and how I could leverage the GPU resources more, instead ending up working on another data-heavy project. I still enjoyed my project and I learned a lot about video models nonetheless but I could have

challenged myself more with some new topic. On the other hand, internships are too short to meaningfully explore a new direction outside of one's comfort zone — at least if publication is the goal. Even while staying in my comfort zone of analyzing shortcuts in VLMs and proposing a benchmark, it was still hard to get it published and felt more rushed than my usual PhD work.

As the Meta internship was coming to an end, I was grappling with big questions, and questioning my direction so far: Was my research fundamental and impactful enough, going beyond just putting out some papers? Did I want to stay in academia? The famous talk by Richard Hamming seems appropriate to link here: You and Your Research. I do not know when exactly my mind shifted from "I want to just get through this PhD and we'll see from there" to "Maybe I will continue in academia". And I also don't remember why exactly I thought interpretability was the right direction. But somewhere around December 2024 these thoughts kept coming up. I had a call with Siva where I said I wanted to actually understand models more deeply, not just do benchmark work or some other modeling work. He pushed back a bit but we mostly just brainstormed and he tried to give his perspective on what faculty search committees are looking for: They want to see that the candidate goes beyond pure analysis or insight for the sake of it, e.g., did your interpretability finding improve models?

The 1.5 years following this chat in December 2024 would be the most fun and scientifically meaningful. I committed to doing something like interpretability on VLMs, without knowing yet what interpretability in detail means or about any of the struggles the field was facing around SAEs, actionability and so on. I really recommend this blog post I wrote in December 2025 and poured my heart into, describing in detail how my pivot into interpretability went: "Better late than never: Getting into interpretability in 2025". I talk about how interpretability forced me to think more scientifically, how great most people in the community are as collaborators, and how overall it was the right decision for my academic career and my excitement for science. I am proud of what came out of this period, which is now Chapter 5. After this experience I feel like a matured

researcher in some way. I have lots of follow-up ideas, I kind of "got the hang of academia", I am mentoring younger students now on VLM interpretability projects. The LatentLens project was also perfectly timed: I was giving lots of talks, some for fun and some as proper job talks, and it felt very natural to be excited as a speaker and showcase a long-term vision when you are just finishing a project you are proud of. Even now, four months after releasing LatentLens, some of that immediate excitement has slowed down and it would be slightly harder to get people hyped as my natural self, if I had to give my job talks now.

Despite how long this section already is, I skipped a lot of meaningful parts of the PhD: The collaborations I was on as a non-first author, running the Mila tea talks, traveling to places, our lab culture at Mila, or all the projects I mentored with undergrads that did not make it to publication. I hope this long story was interesting or helpful to read for some people.

A.3 Staying in academia

I applied for postdocs in a slightly rushed manner. My partner and I had the classic two-body problem and were trying to coordinate our applications (in their case for PhDs), which meant that I had to start sending out my materials by winter 2025/2026. While it was rushed, it turned out to be the perfect timing in most ways, since LatentLens was fresh out of the oven. I applied to various postdoc fellowships, one Research Assistant Professor position and also some stand-alone labs. In the end I chose a postdoc in Boston where I could really focus on interpretability (David Bau) and where my partner had found a great PhD as well. I don't want to share anything prematurely here publicly since it has not started, but lab culture really matters to me and during my job talk I could sense a fantastic one. And I could sense a spirit of doing both fundamental and humanistic science, nurtured by David Bau as the PI.

I really started to enjoy giving talks. They are stressful and often involve travelling[3] but it made me realize how "sharing science" is so much more than

[3] I was quite often sick in that period of both submitting LatentLens to a conference, while

the PDF we put on Arxiv or OpenReview: Impact is much more meaningfully measured in the kind of discussions and ideas your talk, coffee chat or poster might spark. You can, and should, communicate your findings in diverse ways from talks to videos or blog posts. And it is very hard but fruitful to think beyond a single paper and longer-term research question that might take many papers to answer.

A.4 Lessons and reflections

A.4.1 Different research identities

Several times throughout the PhD, my primary research interests, what I deemed interesting papers, or how I pitched myself, changed. There is several distinct "phases" I want to discuss here, you could also call them research identities or, metaphorically, hats I was wearing. In retrospect, each older research identity of myself now seems very narrow-minded and naive. For example, as I was applying and then starting the PhD, I had little admiration for technical modeling or analysis work. The kind that had no philosophical or cognition component. I enjoyed thinking about the big questions, often based on popular science books, podcasts, philosophy papers, or position papers at AI conferences. Now, it is almost the opposite: It is easy to discuss big ideas and define some abstract framework or how one thinks the human mind works. The hard part is all in the nitty-gritty details! I suspect this mindset shift is common in a young person: You start out young and naive in school or undergrad, hoping to change the world, either in science (as discussed here) or industry/politics. Then, you realize that deep familiarity with the subject is required to know how to really make impact.[4] But I still think this period is important, and definitely was for me: Pick a strong opinion and idea and be wrong or naive about it! As long as one is open

also flying to give talks and worrying about the future.

[4]This is actually not always true: Sometimes the most disruptive paradigm shifts in science and technology come from people who either disregard or are simply unaware of the details of the field.

to adjusting it. For me to start something and stick with it, I really have to be convinced. And usually, technical work alone will not do this for me. However, now I am so deep in the work that I enjoy the process so much that I do not necessarily need philosophy to get me going. Interpretability as a field has a lot of philosophy, but it's more like a really nice bonus than something I cling to. After my "philosophy" identity at the start of the PhD, I next adopted a "behavioral analysis and evaluation" mindset. This was also in line with my papers at the time. My argument was roughly that if you want to think about higher-level questions of cognition, the input-output level is much more meaningful than technical modeling work or interpreting certain components of the model. I think there is some truth to it but will not attempt to argue one way or another. I mostly just think it was too simplified of a view and in practice in AI we can derive so many more insights when we also look inside models. However, one opinion from back then I still hold: Too few people deeply study data and individual examples (either from benchmarks or model outputs). So much insight can come from that! Only people who properly value data and benchmarking work put in this effort. During that time, I felt that in practice a lot of evaluation work can be quite shallow, as opposed to answering questions about cognition: "We collected some data, here is how models do". So I was wondering if I should fully commit to being an eval person but make my work more fundamental by going a meta-level higher. "Eval of evals": What trends across many years do we find in benchmarking? How do we go beyond static benchmarks? What makes evals meaningful, without shortcuts or quick saturation? When does Goodhart's Law and the Clever Hans effect come into play? How do we translate some abstract notion of "this is what humans care about" or "this is what we would consider smart behavior" into a concrete benchmark? Moritz Hardt's talk "The emerging science of benchmarks" is great and apparently he also recently released a book: The Emerging Science of Machine Learning Benchmarks. I think I also briefly considered becoming very focused on vision-centric problems, or more broadly world modeling and intuitive physics. The language side of vision-and-language

seems mostly solved, so the video and world modeling side is the obvious next big challenge. So I had all these different identities, summarized as: Philosophy & Cog-sci (2019 to early 2022), AI safety/ethics (briefly around 2022), Behavioral analysis and science of benchmarking (2021 to 2024), world models (briefly in 2024). Nowadays, I am maybe wearing an "interpretability hat" but I wear it loosely and prefer to just tackle interesting problems. More broadly, I like papers that ask a good question, that find something genuinely surprising, do something creative, that questions how we have been doing things in the field, and whose writing is easy to follow — it does not matter too much whether they are about deep cognitive science questions, technical aspects of some training recipe, or interpretability.

A.4.2 Lessons

I will try to condense some lessons here that are otherwise dispersed across the previous sections and my blog posts. As always, all advice comes at the right time for some people and would be outright bad for some others. So choose the ones that apply to you, e.g., keep in mind that advice is often best when it comes from someone who is 1-3 years ahead of you. For example, when I look at undergrads or MSc applying for PhDs these days, it already feels a bit less relatable to me now than it did when I was a 2nd-year PhD student. So here are some things I have concluded during my PhD, I am sure I forgot some and I tried to exclude those that are obvious (e.g., just like anyone else I also recommend having hobbies outside the PhD!):

1. The exact topic is less important than: the people, the lab, the culture, and choosing just an interesting question. The first few you can already judge early on, the last one comes with time and a matured research taste.

2. If you think about pivoting, don't pivot abruptly (unless you really think the old direction does not suit you at all or is out-dated). Instead, pivot one step at a time, so that you don't have to pause for a year just to on-board onto a new field. For example: I had expertise in vision-and-language

models and wanted to get into more fundamental directions (interpretability, more mathy, more theoretical maybe). However, if I had immediately worked on, e.g., theory of Sparse Autoencoders, it would have taken me half a year of just learning about the field, I would have relied much more on guidance from others, and would have faced a higher risk of publishing something nobody cares about. Instead I worked on interpretability of representations in VLMs. While you can never predict how many people care about it, I knew that I and some others would care since I had worked a lot with VLMs. And in the process I also had to absorb what the interp field has been doing in general: so as a next step it would now be easier to work on more theoretical interpretability such as proposing a new type of SAEs, parameter decomposition method. In summary: if you want to pivot to topic A and are currently an expert on topic B, try to work on A+B first before moving to A alone.

3. Apply to positions or give talks when you are just finishing a work you are excited about. Of course, you can't always time this. But sometimes we try to give ourselves more time to be perfectly prepared. But we will never be fully prepared, so we might as well apply or give talks when we are on a high, from a work we feel good about.

4. Similarly, be proactive about giving talks — or, for that matter, writing blog posts, podcasts, and so on. As PhD students we will spend literally months for a paper but think that we don't have the time to, e.g., write a blog post. But the opposite is true: A good blog post (or a long twitter thread, a podcast, a video on YouTube) takes, at maximum, a week to create but the unforeseen effects can often be more impactful than some of your papers. I personally love to see this in a candidate (e.g., when undergrads apply): They have thought about the field or some interesting problem in their own unique way. Not just in the way that academic standards require us to write. It practices your writing (or speaking skills) and makes you

in general less afraid to just put stuff out. And again: Let's say you have already been on so many papers. Do you think quickly putting one more out will give higher chances to land that job? Or instead, what if you spend a few days to write a blog post and share it with people, to reach out to a nearby lab to give a talk, or to start a reading group?

5. Find your people: In my case that meant being slightly less strict with the topics I care about and trying to get more interested in what my labmates do, towards the end of my PhD. If your lab is simply not aligned, you have to go out there and find those people in other departments or universities.

6. Have at least one interpretability-flavored project in your PhD. It will teach you a lot.

7. Read position papers and form some of your own positions. They don't have to be perfect.

8. I collected good advice or generally great blog posts here.

A.5 Original Statement of Purpose when applying

The following is the Statement of Purpose I submitted when applying for PhD programmes in late 2020 (school-specific paragraph omitted).

Towards the end of my undergrad in Computational Linguistics something became more and more apparent to me: the majority of models we currently use in NLP will never reach a "real" language understanding, no matter how much we scale them. Ultimately we will need grounding! These models treat text in isolation, neglecting that language gets its meaning when agents interact in a complex multi-modal environment. Therefore the broader goal of my future scientific contribution is to ground language in perception, action and social interaction to enable language understanding.

This shift of how I see NLP was originally inspired by Experience Grounds Language (Bisk et al., 2020), co-authored by CMU's LTI professor Yonatan Bisk,

that describes a roadmap for Grounded Language research. Bender and Koller (2020) made similar arguments in the best theme paper of ACL 2020 and Language Grounding now has its own track at many NLP conferences, indicating its increased momentum in the NLP community.

An attractive aspect of this subfield is its diverse nature since it can draw from ML, Linguistics, Vision, Robotics or Cognitive Science. Consequently, it ties together previous research experiences of mine from the last several years in NLP, Computer Vision and Robotics:

Research on Symbolic Reasoning in Transformer models: In my thesis, I tried to answer to what extent BERT applies symbolic reasoning during pretraining. The resulting paper *Are Pretrained Language Models Symbolic Reasoners over Knowledge?* will be published at CoNLL 2020, written together with Nora Kassner and my supervisor Hinrich Schütze. Meeting the deadline for CoNLL this summer was hard work: it was only possible because I started my research several months earlier than a typical thesis requires and had a tight collaboration with my co-author Nora. I was involved in every step of the research process, from implementation (creating a synthetic dataset and training BERT from scratch in various ways) to shaping the direction of the project with my own ideas on symbolic reasoning or helping in writing the final paper.

Research on Coreference Resolution in Machine Translation: As a research assistant supervised by Prof. Alexander Fraser, I tested the ability of a context-aware NMT system to resolve coreferences while translating from English to German, e.g. through adversarial attacks. These adversarial attacks ultimately led to a new template test set, which can be used by other NMT researchers. I carefully designed this test set to only allow high scores for models that "truly" solve coreference resolution and not for models that exploit unintended correlations. Similar to my thesis research, I was involved in each step as a shared first-author and wrote several sections of the paper *ContraCAT: Contrastive Coreference Analytical Templates for Machine Translation* that was accepted at COLING 2020.

Industrial research in Computer Vision and Robotics: While having research experience in NLP will be crucial for my PhD, my previous and current experience in Computer Vision and Robotics will complement this effectively:

During the final year of my undergraduate degree, I worked 20 hours per week at a Computer Vision startup as a Deep Learning Engineer. In my main project, I worked on the implementation of an image segmentation system (U-Net architecture) for biological cells. Besides extending my knowledge about Deep Learning and industrial research projects from start to end, I also got to learn how to deal with high-stress situations and tight deadlines with care while working alongside my full-time studies.

I am now using the gap year before my graduate studies to extend my skill set to crucial areas for Grounded Language that I still lack experience in, such as robotics or reinforcement learning. At the time of writing, I am an intern at Micropsi Industries, a Computer Vision and collaborative robots company, where I focus on making the vision system more robust to challenging light settings such as reflections. After initially researching different augmentations, my final solution involves designing a pipeline to automatically record paired image data on the robot which will then be used to train a Conditional GAN. I just started recently, but my work so far already taught me how challenging it is to bridge the gap between our ML systems and real world scenarios.

Why a PhD? Having gained insights into both the world of academia and industry, I can say with confidence that I want to follow the path of the former for the coming years. I am deeply passionate about the fundamental, open-ended, and thorough nature of research, which is especially present in a topic like Grounded Language. By listening to the experiences of countless PhD students and researchers, I got a representative picture of the ups and downs in academia. From my own limited experience so far, I particularly enjoyed the collaboration with other researchers and the scientific inquiries that challenged me and deepened my understanding; however, I also learnt to embrace and even enjoy the struggles such as chaotic experiments and late nights with my

collaborators.

Future research: I am aware that Grounded Language research is still in its early stage and currently is less spectacular than text-only based approaches like GPT-3 or BERT. There are still many fundamental issues to be solved, such as multi-modal models failing to actually use all modalities effectively (e.g. focusing only on text) or tedious data collection and preparation of hardware for embodied systems in the real world. More specifically, I can see myself conducting Grounded Language research in three ways:

- From a cognitive and linguistic side, I would be interested in modeling how humans acquire language (e.g. similar to Emin Orhan et al. (2020) who studied the vision aspect of this) or simulating how language emerges between non-human agents (Lazaridou et al., 2018). I also believe that it is crucial to develop persistent agents that acquire grounded temporal commonsense (Zhou et al., 2019) and would intuitively know that e.g. "going on a vacation" will take at least several days. Such intuition could be learnt by watching videos of people's daily activities, such as in the Epic Kitchens Dataset (Damen et al., 2020).

- I would like to make progress on different grounded benchmarks such as visual (commonsense) reasoning or visual-and-language navigation (e.g. Yonatan Bisk's ALFRED benchmark (Shridhar et al., 2020)).

- Finally, I would also like to put some focus on creating better data sets, environments or simulations that will enable a wide range of Grounded Language research. More specifically, I recently became interested in investigating whether big multi-player games such as Sims or WoW could be useful environments for grounding, especially with a focus on social interaction.

Multi-modal models will play a role in my research as useful tools, but are not equivalent to Language Grounding research: A primary goal of Language

Grounding is to understand the nature of language and its role in intelligent behaviour, whereas pure multi-modal approaches in NLP tend to use other modalities to achieve better performance on certain tasks. In short, Language Grounding encompasses more than putting together language and vision models, and I would like my research to reflect that.

B

Appendix for Chapter 2

B.1 Length Distribution of the Image Descriptions

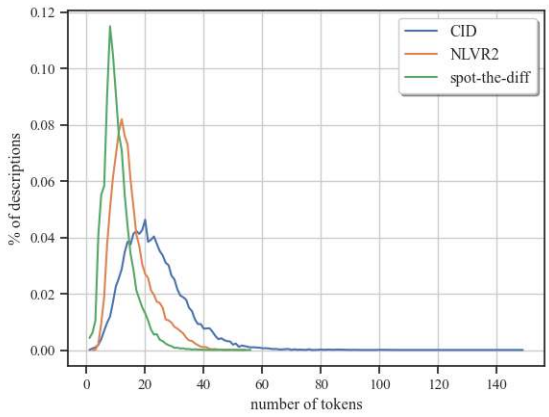


Figure B.1: Distribution of the number of tokens across contextual descriptions in IMAGECODE.

B.2 Criteria for Selecting Annotators

We keep data quality high through entry requirements (English speaking country, over 98% approval rate, etc.), qualification test, whitelisting workers and manually inspecting data. Most importantly our two-stage setup also allowed us to automate monitoring data quality as we could measure the description and retrieval accuracy of workers and only whitelisted those with high accuracy. We paid 0.25$ per description and 0.1$ per retrieval.

B.3 Annotator Bias

The majority of descriptions in our test and validation split come from workers who did not work on the training set in order to avoid annotation bias. Our validation set contains 502 descriptions from workers "seen" from the training set and 1,800 description from "unseen" workers. In Table B.1 we can see that models perform slightly better on seen workers across our CLIP model variants.

	seen workers	unseen workers
FINE-TUNING		
CLIP	**23.9**	23.8
+CONTEXTBATCH	**34.5**	29.0
+CONTEXTMODULE	**33.3**	29.2
+TEMPORALEMBEDDINGS	**32.1**	30.8

Table B.1: Performance (accuracy) on two subsets of the distinct validation split: seen workers (workers who also produced description on the train split) and unseen workers (who only worked on the test and validation data).

B.4 Crowdsourcing Interface

Our AMT interface for the description task can be seen in Figure B.2. The retriever interface looks conceptually similar, with a select-button for each image. Note that workers see images almost in almost half of full-screen (opposed to the shown examples in this PDF) and can quickly go back and forth between consecutive frames with arrow-keys, making it significantly easier to spot and compare nuanced changes.

Figure B.2: AMT interface for the describer task.

B.5 Validation performance

	all	video	static
ZERO-SHOT			
CLIP	21.8	14.9	51.6
FINE-TUNING			
CLIP	23.4	17.3	50.2
+CONTEXTBATCH	29.7	21.1	67.2
+CONTEXTMODULE	29.9	21.4	**67.2**
+TEMPORALEMBEDDINGS	**30.6**	**22.3**	67.0
ZERO-SHOT			
UNITER	19.8	13.6	42.9
FINE-TUNING			
UNITER	23.8	17.5	51.2
+CONTEXTBATCH	25.5	19.3	52.3
+CONTEXTMODULE	24.8	18.9	50.7
+TEMPORALEMBEDDINGS	**26.0**	**19.9**	**52.8**
ZERO-SHOT			
ViLBERT	18.5	14.0	37.9
FINE-TUNING			
ViLBERT	21.9	16.1	46.7
+CONTEXTBATCH	22.9	18.1	43.5
+CONTEXTMODULE	23.5	18.9	43.5
+TEMPORALEMBEDDINGS	**25.1**	**19.4**	**49.5**

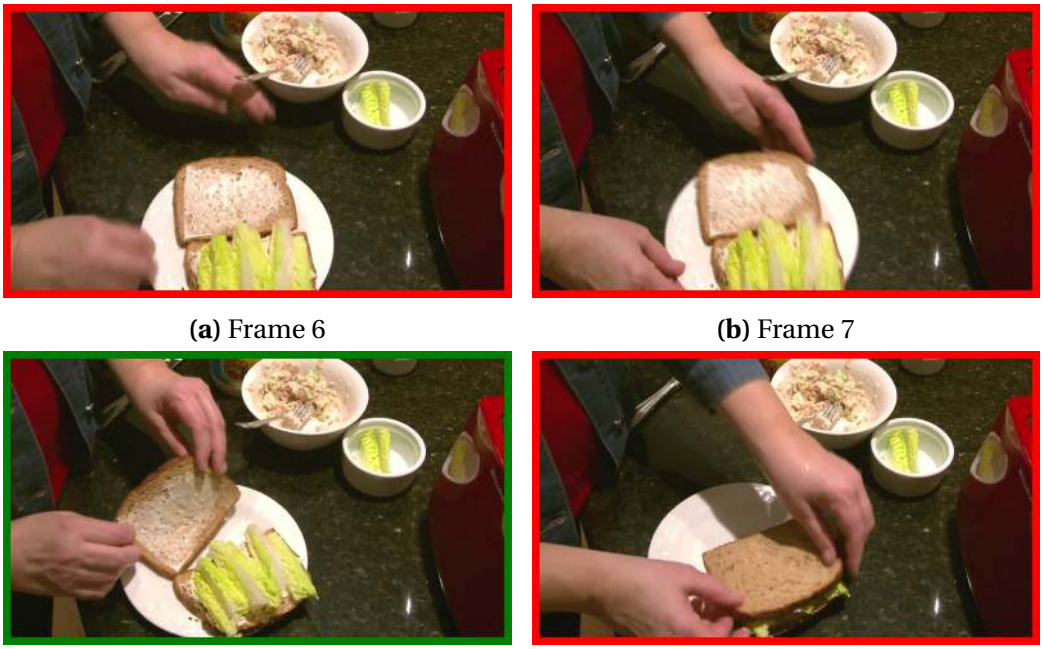
Table B.2: Performance (validation accuracy) on IMAGECODE across two training regimes (zero-shot and fine-tuning), three models (CLIP, UNITER, ViLBERT) and 4 model variants. We report separate figures for all the examples and two disjoint subsets: video frames and static pictures.

B.6 Additional Hyper-parameters

The Transformer consists of 2 layers in CLIP variants and 4/5 layers in the ViLBERT/UNITER variants, both employing gelu activation. The learning rate for the fine-tuning of the Transformer and linear heads is $2 \cdot 10^{-6}$ for the CLIP +CONTEXTMODULE, 10^{-4} for CLIP +TEMPORALEMBEDDINGS, $2 \cdot 10^{-5}$ for both ViLBERT variants, and $6 \cdot 10^{-6}$ for both UNITER variants. We use the Volta-framework (Bugliarello et al., 2021) for the standardized ViLBERT and UNITER model.

B.7 Examples from IMAGECODE for all phenomena

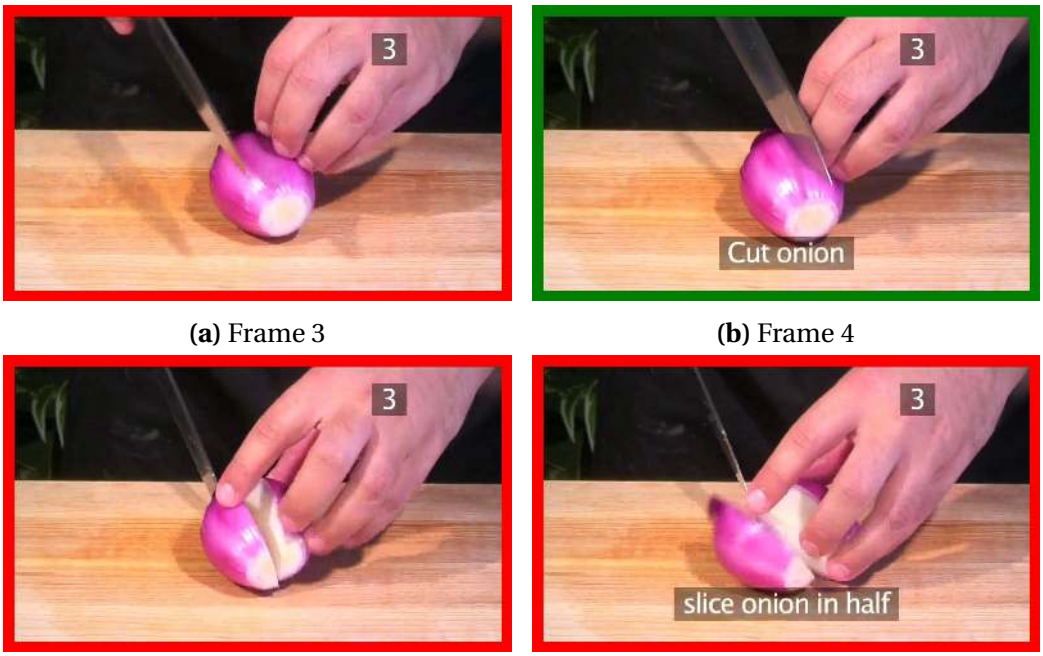
For each phenomenon we provide 1 example and a definition we used for annotation purposes. Since most examples contain more than one phenomenon, some phenomena will be effectively showcased several times. Note that we picked examples that are relatively easy to understand and spot differences in.



(a) Frame 6 **(b)** Frame 7

(c) Frame 8 **(d)** Frame 9

Figure B.3: Example of **Context**: *"Both hands are on the piece of bread closest to the person."* Note: This is contextual since since without any context of other images, the description is also literally true for Frame 9. A model might even score it higher since the direct visual appearance is closer to typical bread. Definition: To understand the description, a listener has to consider other images and/or the speakers intention of describing only one of the images. In line with Grice's maxim of quality, a description is contextual if it is literally true for several images but we know it was intended for only one image. A description is also contextual if an objects cannot clearly be identified in the target image directly but only through cross-referencing other images.



(a) Frame 3 **(b)** Frame 4

(c) Frame 5 **(d)** Frame 6

Figure B.4: Example of **negation**: *"The knife is most centrally placed to insert into the onion **without** having fully cut deeply into it yet."* Definition: Explicit linguistic negation ("not", "unseen", "non-") or negation quantifiers ("no person").

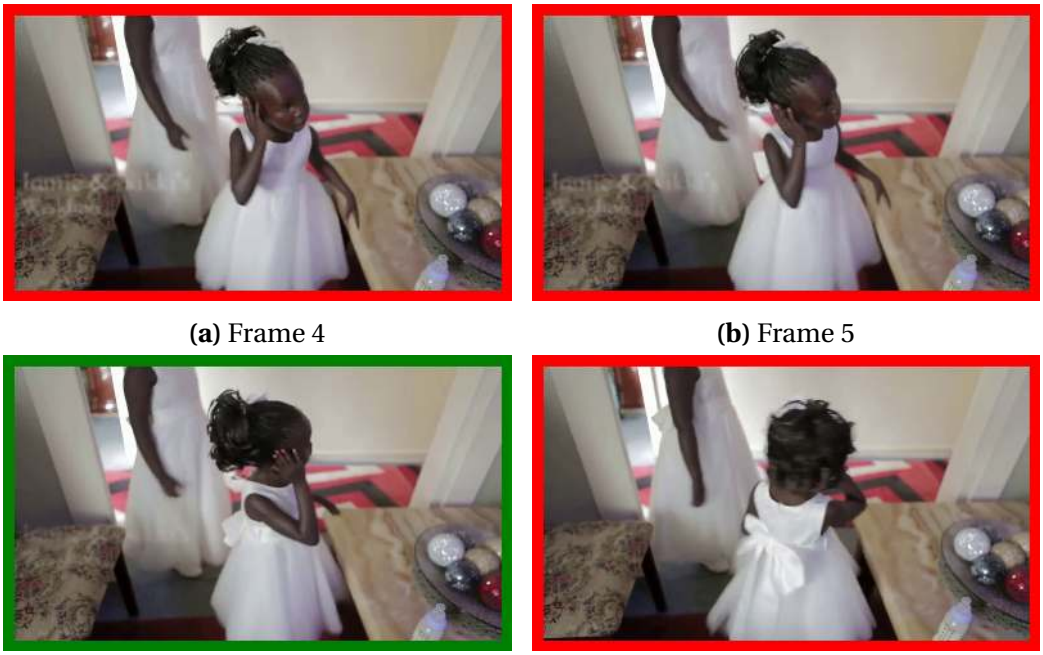
(a) Image 1

(b) Image 3

(c) Image 8

(d) Image 9

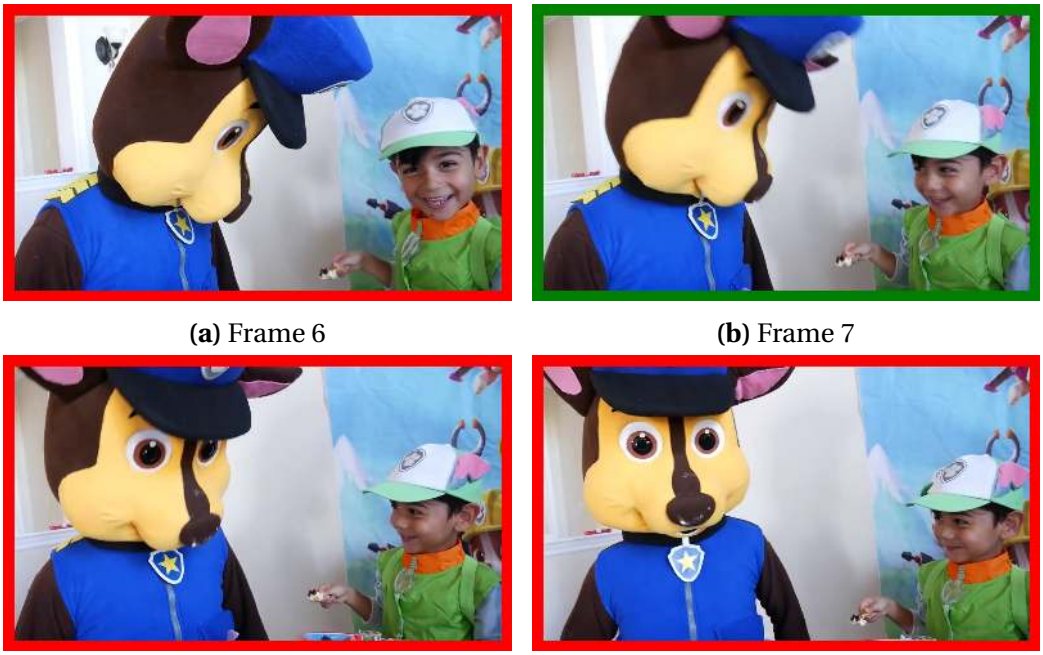
Figure B.5: Example of **quantifiers/quantities**: *"A yellow **3 way** traffic light with a green arrow on the side facing closest to the camera"* Definition: We annotate for quantifiers (most, every, no, several,...) and absolute quantities ("five") as well as relative quantities (ratios like " a third of his hand").



(a) Frame 4 **(b)** Frame 5

(c) Frame 6 **(d)** Frame 7

Figure B.6: Example of **spatial relations/reasoning**: *"The small girl **in front** is looking **directly to the right** with her **right hand on the side** of her face."* Definition: Any relations or adjectives regarding space. Examples: "in the top left corner", "left to the chair", but also camera perspective, or body orientation ("turned towards...")



(a) Frame 6 **(b)** Frame 7

(c) Frame 8 **(d)** Frame 9

Figure B.7: Example of **temporality**: *"A smiling boy **just begins** to look towards the dog."* Definition: While most examples based on video frames implicitly require some temporal knowledge, we focus on explicit textual mentions of 1) temporal markers ("after", "during", "about to", etc) and 2) temporal verbs ("beginning to", "end to").

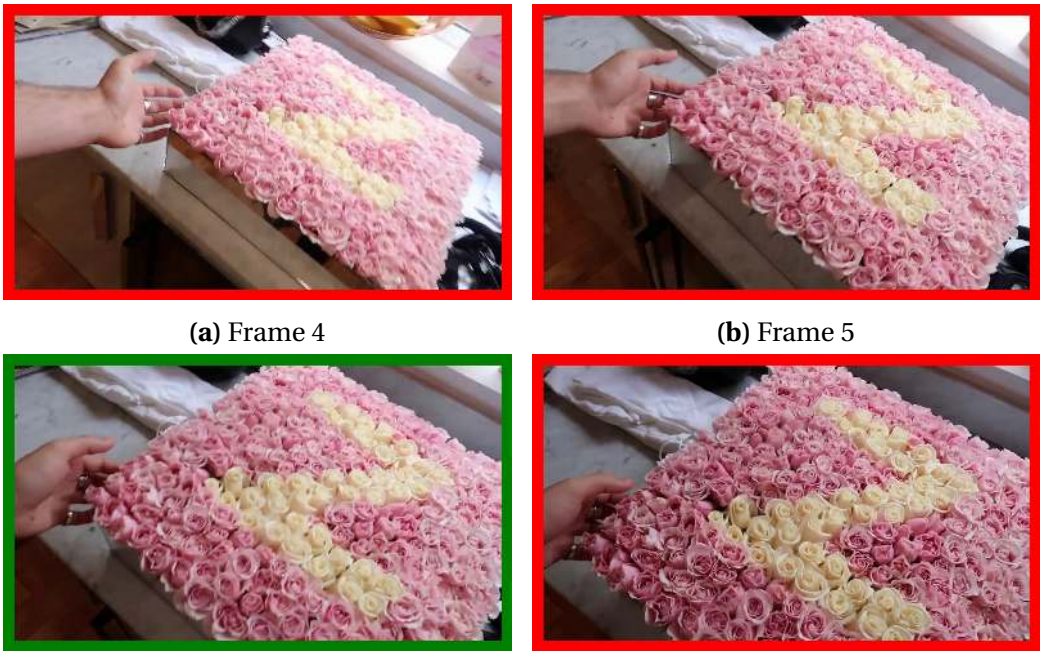
(a) Frame 7

(b) Frame 8

(c) Frame 9

(d) Frame 10

Figure B.8: Example of **visibility/occlusion**: *" The tire is directly **on top of** the person's right shoe and you can **just barely see** fingers at the top. "* Definition: A description that mentions objects/people being occluded, (partially) out of frame, or in the process of leaving the frame.



(a) Frame 4 **(b)** Frame 5

(c) Frame 6 **(d)** Frame 7

Figure B.9: Example of **nuances** (we marked small details with red/green rectangles): *" The person's palm is towards us and touching the left bottom corner of the cake. There is a* ***small amount of dark space between the right bottom corner of the photo and the edge of the cake****. "* Definition: Minor details, that are either a) not salient at all and would usually be left unmentioned and/or b) language reference is grounded on a small patch of pixels. Note that this phenomena is often linked with very minimally contrastive images.

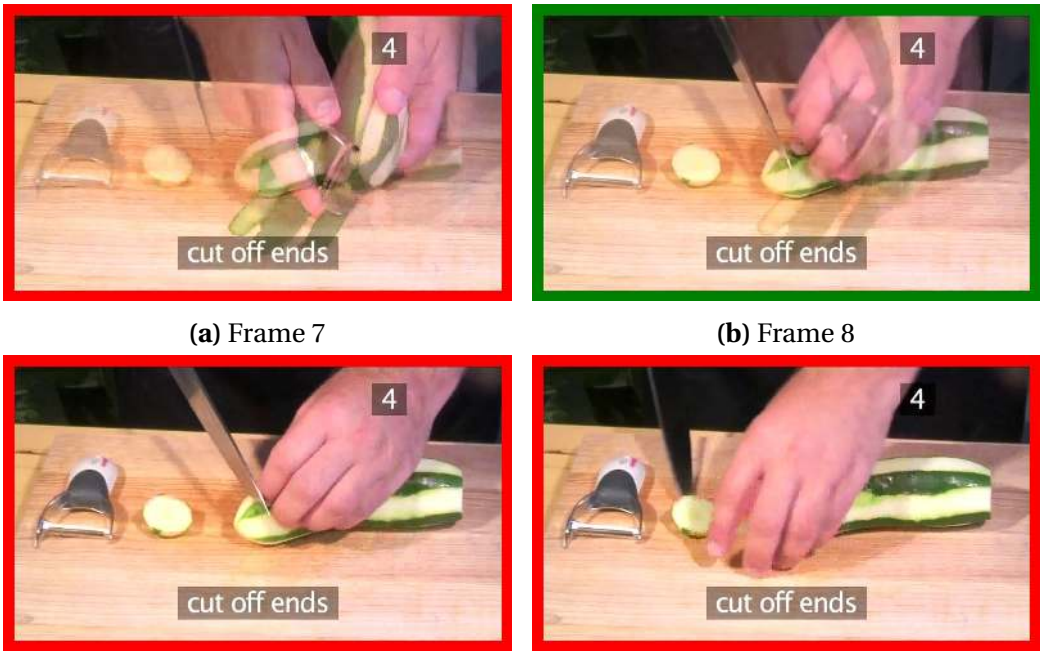
(a) Image 3

(b) Image 5

(c) Image 6

(d) Image 10

Figure B.10: Example of **coreference**: *" A woman with a white background smiles at the camera. Most of **her** body is visible. **She** is wearing a black outfit. "* Definition: Linguistic coreference.



(a) Frame 7 **(b)** Frame 8

(c) Frame 9 **(d)** Frame 10

Figure B.11: Example of **meta properties**: *" The cucumber is just to be cut into, you can see a* ***transparent image covering the image****."* Definition: Descriptions that mention aspects that stem from the way the photo/video was taken: two overlayed images (when a video transitions), black-and-white, blurriness, brightness.

C

Appendix for Chapter 3

C.1 Dataset Release

In this section we document the details of our dataset drawing from existing frameworks such as Datasheets for Datasets (Gebru et al., 2021).

1. **Data and code**: `https://github.com/McGill-NLP/AURORA`
2. We provide a **Datasheet** (Gebru et al., 2021) for our **AURORA** and **AURORA**-BENCH data as a Markdown file: `https://github.com/McGill-NLP/AURORA/blob/main/datasheet.md`, as well as at the end of our appendix: Section C.14.
3. **Hosting Plan**: As described in our instructions on the GitHub repository, we host our data on Zenodo and intend to put it on Huggingface soon after submission

4. **Licensing**: We release all our data (**AURORA**, **AURORA**-BENCH) under the **MIT License for easy access to other researchers**. The license allows users to share and adapt the dataset for any purpose, even commercially, as long as appropriate credit is given and any changes made are indicated. The datasets we build upon or directly include in our collection of data have the following licenses: MagicBrush (CC 4.0), Action Genome (MIT), Something Something (see `https://developer.qualcomm.com/software/ai-datasets/something-something` for their terms of use, more restrictive than MIT or CC 4.0).

5. **Consent & Privacy**: We work with Amazon Mechanical Turk crowdworkers who did not share any private information. Two of the datasets we build on top of, Action Genome (Ji et al., 2020) and Something Something (Goyal et al., 2017a), asked humans to film videos at home doing daily activities. Action Genome builds on top of Charades (Sigurdsson et al., 2016), and we did not find any mention in their paper discussing privacy or consent: Workers were recruited on AMT. The situation is similar for Something Something but we would assume that workers were told and aware that their data is going into a public research dataset. This is also officially part of the agreement when becoming a worker on AMT.

C.2 Broader Dataset Discussion

C.2.1 Limitations

We acknowledge that these models are not yet mature to robustly perform the edits we study in this paper: Even in simpler setups such as Kubric, WhatsUp or CLEVR we still observe some failures, and on messy data where humans perform actions fully correct edits are rare, as reflected in the human ratings (Table 3.2. Even on the more established object-centric and global we still identify many failures, arguably more than in the more mainstream text-to-image generation.

Specifically we identify the following failure modes during many manual :

Models pick up artifacts if something is overrepresented in **A**U**RO**R**A**: It might over-generate hands or people due to many such examples in the video-frame-based data. Similarly, the model sometimes falls back to textures and shapes from Kubric when common Kubric phrases (i.e. cups) are mentioned. While our model has learned spatial relations, it still struggles with "truly moving" an object: Sometimes the original objects is kept at its place while a new one is added elsewhere, resulting in too many objects. Often, the properties of the object (i.e. size or texture) also change, and can become more "Kubric-like" after being moved. Finally, while many Kubric edits were successfully performed on its IID test examples, *swapping the position* of two objects was not.

C.2.2 Ethical and Societal Discussion

While we envision robotics and other planning tasks as an exciting new application of models that can perform action and reasoning-centric edits, there are several potential harms specific to the broader editing task. The main one is editing images in harmful or privacy-intrusing ways. These harmful edits are not in our training data, but the underlying Stable Diffusion model was trained on them and thus sometimes generates various harmful or biased images (Birhane et al., 2021; Luccioni et al., 2023). We found the efforts of MagicBrush (Zhang et al., 2024) helpful, a dataset we include in our **A**U**RO**R**A** dataset, such as minimizing these problems in their collection as described in their appendix.

On the crowdsourcing side, we ensured fair pay and treatment with compensation far above the minimum wage in the US, see Section C.4. On top of pay, workers gave us feedback several times that they appreciated the feedback, respect for their work and detailed communication.

C.3 Examples of edit typology

In this section, we illustrate our typology of edit skills from Section 3.2. For more examples for each dataset (not skill!) see Section C.10.

Figure C.1: Object-centric edit example: *Can we have a wooden table?*

Figure C.2: Global edit example: *Make it a picnic*

Figure C.3: Action-centric edit example: *Make the man open the refrigerator*

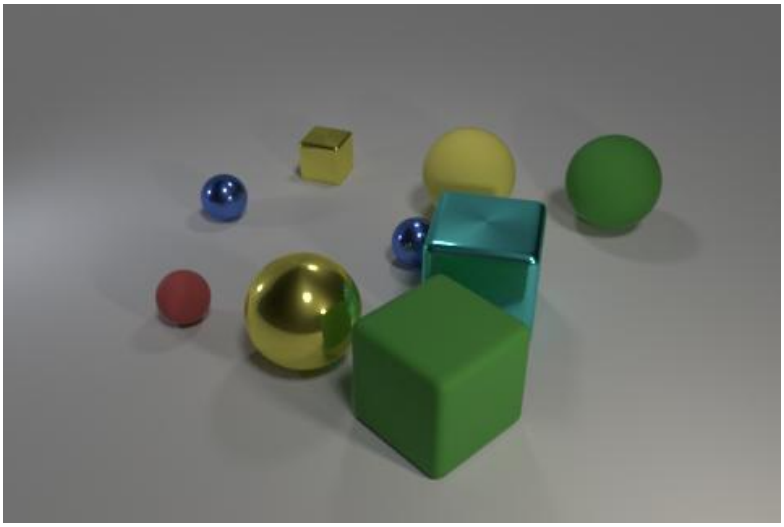


Figure C.4: Reasoning-centric edit example (spatial, referring expression): *Move the green sphere in front of the red small sphere*

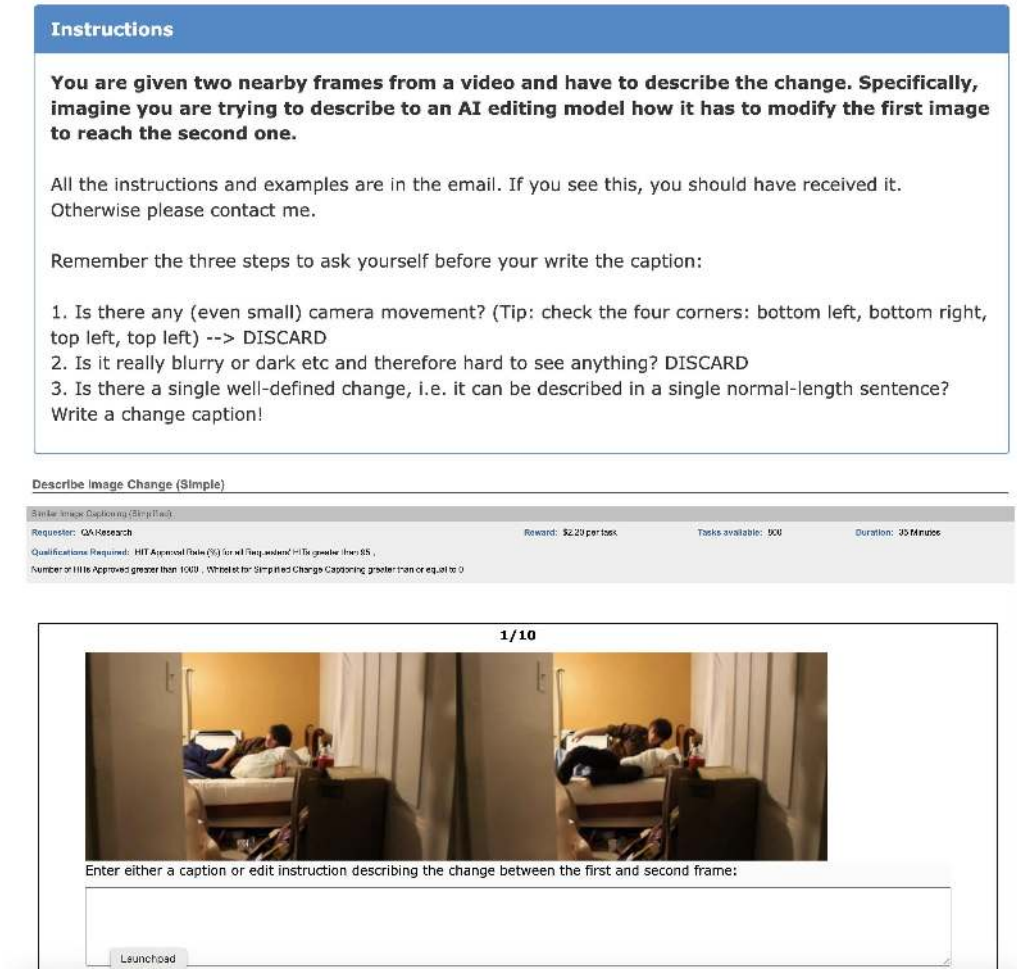
Figure C.5: Viewpoint edit example (from a dataset we later discarded, see Section C.13): *Move the camera left a bit to capture the first person on the left well*

C.4 Human annotation guidelines and details

We work with a smaller group of expert annotators on AMT (who we individually contact via e-mail) after an initial screening during pilot runs. Seven workers first worked on describing changes between nearby video frames as edit prompts, and later on three of those workers for the human judgement. We found that (unsurprisingly) paying well and regular detailed communication led to the very clean results and resulted in, to put it directly, amazing feedback working with us as data collectors: We paid 0.22 USD for the captioning, and 0.20 USD for the human judgement. We estimated this to be significantly above 10 USD/hour, and probably closer to 15 USD/hour. Below we describe the instructions and communication with workers for each task.

C.4.1 Crowdsourcing edit instructions

Our instructions on the AMT interface looks as follows:



Since most of our detailed instructions and back-and-forth feedback happened via e-mail, we also provide screenshots from our e-mails:

The task:
You are still describing the difference between two images. However, two things are different now.
We want to keep it a lot simpler: So we **discard all examples where the camera moves or where more than one change happens at once.** We also discard everything that is dark/blurry.
Specifically, you go through this thought process:

1. Is the camera moving at all? Write "DISCARD" (see tips below on how to quickly see whether the camera moved)
2. Is the image very blurry, very dark etc? Write "DISCARD"
3. Is there only 1 or maximally two changes happening and can you **describe them in simple short language?** If yes, write a short edit instruction! But if it is hard to describe in language and takes a lot of words because there are many changes or unusual changes, write "DISCARD" instead.

I will now give you 10 examples and show you how I want you to annotate it:

1. Is the camera moving at all? We care about even small camera movements. My tip: The best way to figure this out is by looking at the four corners and edges. In this case we can look at the bottom left corner: There is no camera movement :) That's good. So we can move to the next decision.
2. Is the image dark or blurry? Not really, so that's good too.
3. Can we describe the image in simple clear language with a single change happening? Yes :) So that means we **do not discard this example** and describe the change: **"The man lifts up the milk container and drinks it"**

C.4.2 Evaluation of model outputs

For the human judgement comparing two model outputs, we mainly released our instructions as a video. Our inter-annotator agreement was a Fleiss-Kappa score of 0.626. We asked people to pay attention whether the semantics of the prompt was followed and not so much aesthetics:

Hello!

It is time to get things started.
Here are the steps:

1. https://drive.google.com/file/d/1TZu8wRJdo2IgwGdnEvxO0UyEsK0EKyJI/view?usp=sharing
Carefully watch this video, pause sometimes to test your intuition if you would judge it like I do, before continuing to hear me explain it.
These are the main instructions you will get!

2. **Respond after you have watched it**, and ask questions if there are some at this point.

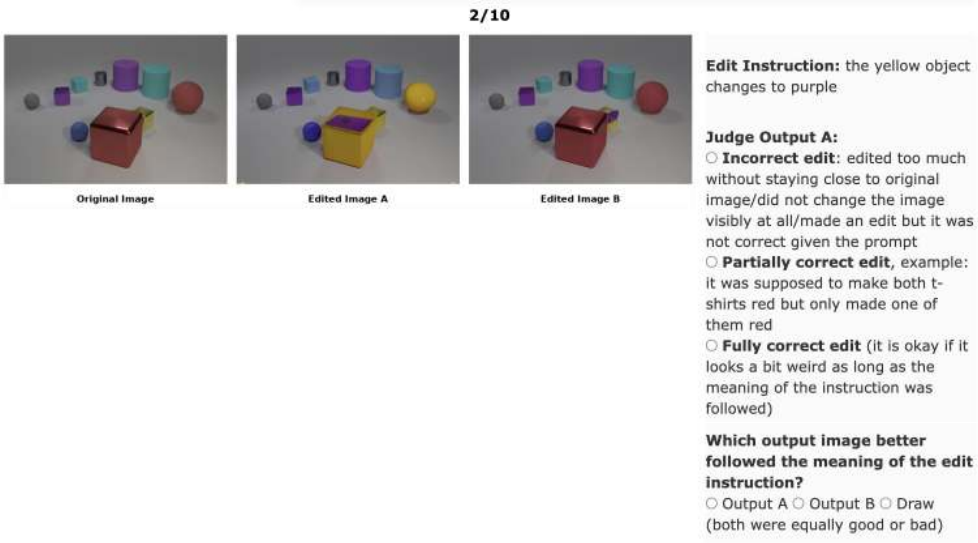
3. Once people have responded, I will release the first 50 as a test run. **For the first 5 examples (not those I showed in the video), please in addition to annotating also write your reasoning here in the email for me to make sure you understood the task.**

4. Then I will release the rest (1K for each of you).

After this we will see if there is more to do in a week!

Thanks and best wishes,

Our AMT interface looked as follows:



C.5 Data Curation Details

Something-Something-Edit: We select the first and last frame of most videos from the Something Something v2 dataset (Goyal et al., 2017a) (most, since for some the ffmpeg software somehow couldn't retrieve the last or nth-last frame). Next, we manually went over the 174 templates that were used to create the original captions such as: "Dropping something onto something" or "Folding something", where *something* would be replaced by the respective objects. We then find certain keywords that rarely lead to useful changes and filter them out. Specifically:

{'pretend', 'holding', 'fail', 'roll', 'then letting it',
'until it falls down', 'spin', 'nothing happens', 'show',
'falling like', 'squeezing', 'throwing', 'slightly'}

Action-Genome-Edit: For this, we first take the original frame pairs from (Wang et al., 2023a) and primarily filter them for similarity, which we found to be a good proxy of camera movement. After some tuning, we settle on a cosine similarity between 0.1 and 0.4 of the two CLIP visual embeddings, allowing us to provide crowdworkers with 20K or more images; we ultimately only use 15K for this. Next, we ask humans to describe these changes and discard anything

with slight camera changes, see Section C.4.1 for more details. After the human filtering phase, Action-Genome-Edit contains 11K examples.

Kubric: These are the procedures used to generate new data for each change type: (1) *Location* changes contain one or two objects moved. They can be relative ("Move O_1 to the left of O_2"), absolute ("Move O further left"), attract/repel ("Move O_1 and O_2 closer together/further apart") or swapping ("Swap the positions of O_1 and O_2") movements. (2) *Rotation* changes rotate an object around the X, Y or Z axis at 90°or 180°. Specifically, we define four rotations: turn O around (Z, 180°), turn O 90 degrees (Z, 90°), make O fall over (X/Y, 90°), and flip O upside down (X/Y, 180°). Some objects are either invariant to certain rotations (i.e. a bowl when turning it around: 180°, Z axis) or the edit is physically implausible (i.e. only vertically long objects such as cups make sense to "fall down": 90°X/Y axis). So we categorized the 213 objects into "can fall", "round" and "fully invariant" before data synthesis. (3) *Count* changes require adding or removing n instances of some object. (4) *Attributes* changes vary the size, shape or color of some object. Examples for each edit in Section C.10.

C.6 Training Details

We follow the common InstructPix2Pix setup regarding architecture and training, and rely on their code implementation (Brooks et al., 2023). Specifically, we finetune with a batchsize of 32 and a learning rate of 5.0e-05. We also experimented with a higher resolution of 512 but saw no clear benefits (however further experiments could yield different results). We turn off the flipping augmentation for datapoints containing the word *left* or *right*, which was not an issue in previous edit training setups but would be a major issue with our focus on spatial reasoning.

Most time was spent on finding the right ratio for mixing the dataset: Our final first trains on MagicBrush alone for 13K steps and then on the full mix for 42K steps. We upsample the datasets such that they are all sampled roughly the same amount, i.e. MagicBrush and Action-Genome-Edit are multiplied by a

factor of 15 while Something-Something-Edit and Kubric remain with a factor of 1. Upsampling action and reasoning-centric edits too much would result in model generations with sometimes too many or random changes. While it might've led to slightly higher numbers on those tasks, the performance on the tradtional edit tasks degraded significantly.

We had access to a total of eight NVIDIA RTX A6000 and trained models on two of them, parallelizing 4 training runs occasionally. Our training run used for the final model would run for 16 hours. We did not keep track of total compute we spent but there were many attempts at combining the datasets in different ways or training on dataset that were ultimately discarded.

C.7 Extended Tables

Main tables (Table 3.2 and Table 3.3a) but with additional confidence statistics (i.e. standard errors).

Table C.1: Extended Table of **Human Judgment** of semantic editing success on **AURORA**-BENCH tasks. Humans were asked to rate the edit success from none (0), partial (50) to full (100). We show standard error (SE). Overall score is "balanced": we average each skill first, and then take the average of those 4 numbers.

Model	Obj./Attr.	Action, Human-Object-Interaction			Reasoning			Global	
	Magic Brush	Action-Genome	Something Something	Epic Kitchen	WhatsUp	Kubric	CLEVR	Emu-Global	Overall Score
GenHowTo	18.0 ± 4.7	8.0 ± 2.6	8.7 ± 2.9	17.7 ± 4.6	4.3 ± 1.7	0.7 ± 0.5	2.0 ± 1.4	11.3 ± 3.1	10.8 ± 1.2
MGIE	36.0 ± 7.2	7.0 ± 2.4	11.3 ± 3.7	5.0 ± 1.9	6.0 ± 1.8	6.7 ± 2.8	16.0 ± 3.9	36.5 ± 6.5	22.5 ± 1.8
InstructPix2Pix	31.3 ± 6.8	13.3 ± 3.5	12.3 ± 4.2	4.3 ± 2.1	0.7 ± 0.7	5.7 ± 2.1	14.7 ± 3.6	33.7 ± 6.8	20.5 ± 1.8
MagicBrush	61.7 ± 9.7	16.3 ± 4.4	17.0 ± 4.7	12.0 ± 3.5	3.0 ± 1.7	9.3 ± 2.9	22.0 ± 4.5	42.3 ± 7.7	32.6 ± 2.4
Ours	60.5 ± 9.6	35.6 ± 6.6	31.8 ± 6.5	14.3 ± 4.0	27.3 ± 5.5	59.6 ± 9.3	46.1 ± 8.1	33.0 ± 6.5	41.3 ± 2.7

Table C.2: Extended table for DiscEdit performance with standard error (SE).

Model	WhatsUp	Something	AG	Kubric	CLEVR
MagicBrush	0.472 ± 0.012	0.371 ± 0.043	0.477 ± 0.043	0.392 ± 0.045	0.400 ± 0.045
AURORA	0.565 ± 0.009	0.548 ± 0.045	0.583 ± 0.043	0.592 ± 0.045	0.450 ± 0.045

C.8 Details of *DiscEdit*

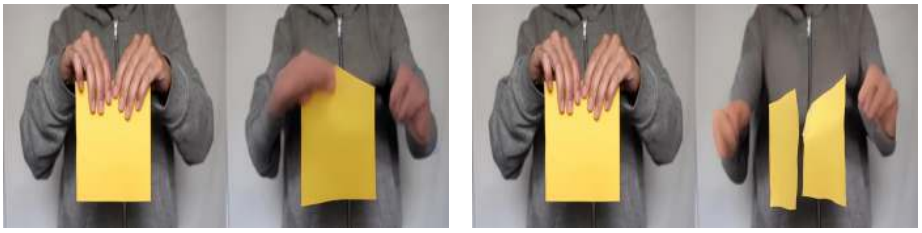
Our *DiscEdit* metric takes an images and two similar prompts $t_{nochange}$ and t_{change}, where the model is expected to not change anything given $t_{nochange}$ but change something (i.e. normal edit) with t_{change}. Practically, one can take an existing image-edit pair dataset and either automatically or manually come up with ”no-change“ prompts.

Here we illustrate this process for some of **AURORA**-BENCH examples. Note that some edits do not have a straightforward associated ”no-change“, most notably adding objects which are many MagicBrush edits. For example, ”add a squirrel next to the lamp“ does not have any equivalent ”no-change“. Let's imagine there is already a dog next to the lamp; so could we have ”add a dog next to the lamp“ as $t_{nochange}$? No, since it would be a correct model response to then add a second dog. So here are examples for several **AURORA**-BENCH tasks, with the goal of illustrating the point of changing *more or less relatively speaking*, not to show cherry-picked perfect examples - WhatsUp, Something Something, AG, Kubric, CLEVR:

(a) No-change prompt: *Move the bowl under the table*

(b) Change prompt: *Move the bowl on the table*

Figure C.6: WhatsUp



(a) No-change prompt: *Keep the paper intact*

(b) Change prompt: *Tear the paper into two pieces*

Figure C.7: Something Something

(a) No-change prompt: *Make him face to the right*

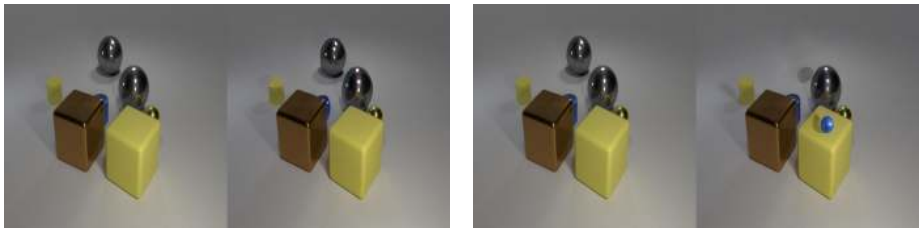
(b) Change prompt: *Make him face towards the camera*

Figure C.8: Action Genome

(a) No-change prompt: *Remove all yellow objects*

(b) Change prompt: *Remove all green objects*

Figure C.9: Kubric



(a) No-change prompt: *Move the blue small ball behind the brown cube*

(b) Change prompt: *Move the blue small ball in front of the brown cube*

Figure C.10: CLEVR

C.8.1 Implementation details

For each triplet of (image, change-prompt, no-change-prompt), we start with the same initial noisy latent for both minimally different prompts to reduce variance, a common procedure in using text-to-image models as discriminators (Krojer et al., 2023b; Li et al., 2023a). To have a larger sample size, and thus reduce variance further, we also sample four different noisy latents per triplet and treat it is as separate examples when computing the accuracy.

We end up using 30 examples for Something-Something-Edit, Action-Genome-Edit, Kubric and CLEVR, and all 800 examples from WhatsUp. WhatsUp was the only task that needed no manual writing or filtering of minimally different prompts due to its well-defined spatial movement setup.

C.9 Sample generations from our evaluation

C.9.1 Samples used for human judgement

We show four randomly sampled outputs from our **A**U**RO**R**A** model on each of the eight **A**U**RO**R**A**-BENCH tasks.

(a) Prompt: *change the bright lime green curtains to white curtains*

(b) Prompt: *Let the blanket be longer and hang over the bed.*

(c) Prompt: *Make the hydrant all white*

(d) Prompt: *Change the banana to a microphone.*

Figure C.11: MagicBrush

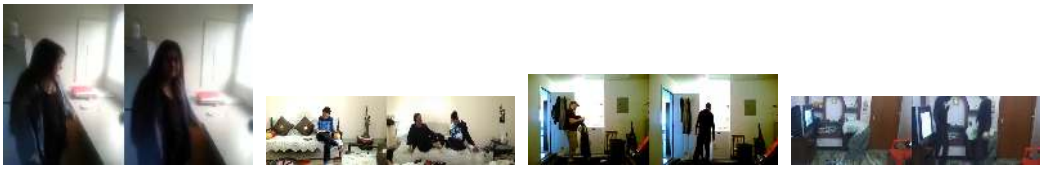
(a) Prompt: *Flip the bottle upside down*

(b) Prompt: *Putting egg into the bowl*

(c) Prompt: *Moving cup away from pen*

(d) Prompt: *Lift her hands up to show a piece of paper*

Figure C.12: Something-Something-Edit



(a) Prompt: *Make her face fully visible looking at the camera*

(b) Prompt: *Make the man put down the book on the sofa*

(c) Prompt: *Make him walk further towards the right*

(d) Prompt: *Make them stand up*

Figure C.13: Action-Genome-Edit

(a) Prompt: *place the black hat on the right hand of the yellow nesquik chocolate powder canister*

(b) Prompt: *convert the white keyboard into a black keyboard*

(c) Prompt: *add 1 white keyboard to the platform*

(d) Prompt: *turn the blue shoe into a white running shoe*

Figure C.14: Kubric-Edit

(a) Prompt: *Move the kettle to the right of the chair*

(b) Prompt: *Move the dog to the right of the table*

(c) Prompt: *Move the bowl under the table*

(d) Prompt: *Move the book behind the tape*

Figure C.15: WhatsUp

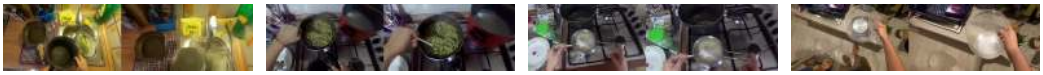
(a) Prompt: *the cylinder becomes gray*

(b) Prompt: *the big metal sphere becomes gray*

(c) Prompt: *make the big green rubber cylinder turn purple*

(d) Prompt: *move the red cylinder to the right of the gray block*

Figure C.16: CLEVR



(a) Prompt: *Grab the soap bottle on the right with the hand*

(b) Prompt: *Pull one noodle out of the pot with the fork*

(c) Prompt: *Move the right hand to one of the oven knobs*

(d) Prompt: *Drop the transparent large bowl onto the floor*

Figure C.17: EpicKitchen

(a) Prompt: *Make this image look like the Simpsons Cartoon.*

(b) Prompt: *Change the image to be taking at night.*

(c) Prompt: *Make it look like a renaissance painting.*

(d) Prompt: *Give this image a look inspired by Michelangelo's "David".*

Figure C.18: Emu

C.9.2 For DiscEdit

See Section C.8

C.10 Random (=non-cherry picked) Samples from existing and contributed training datasets

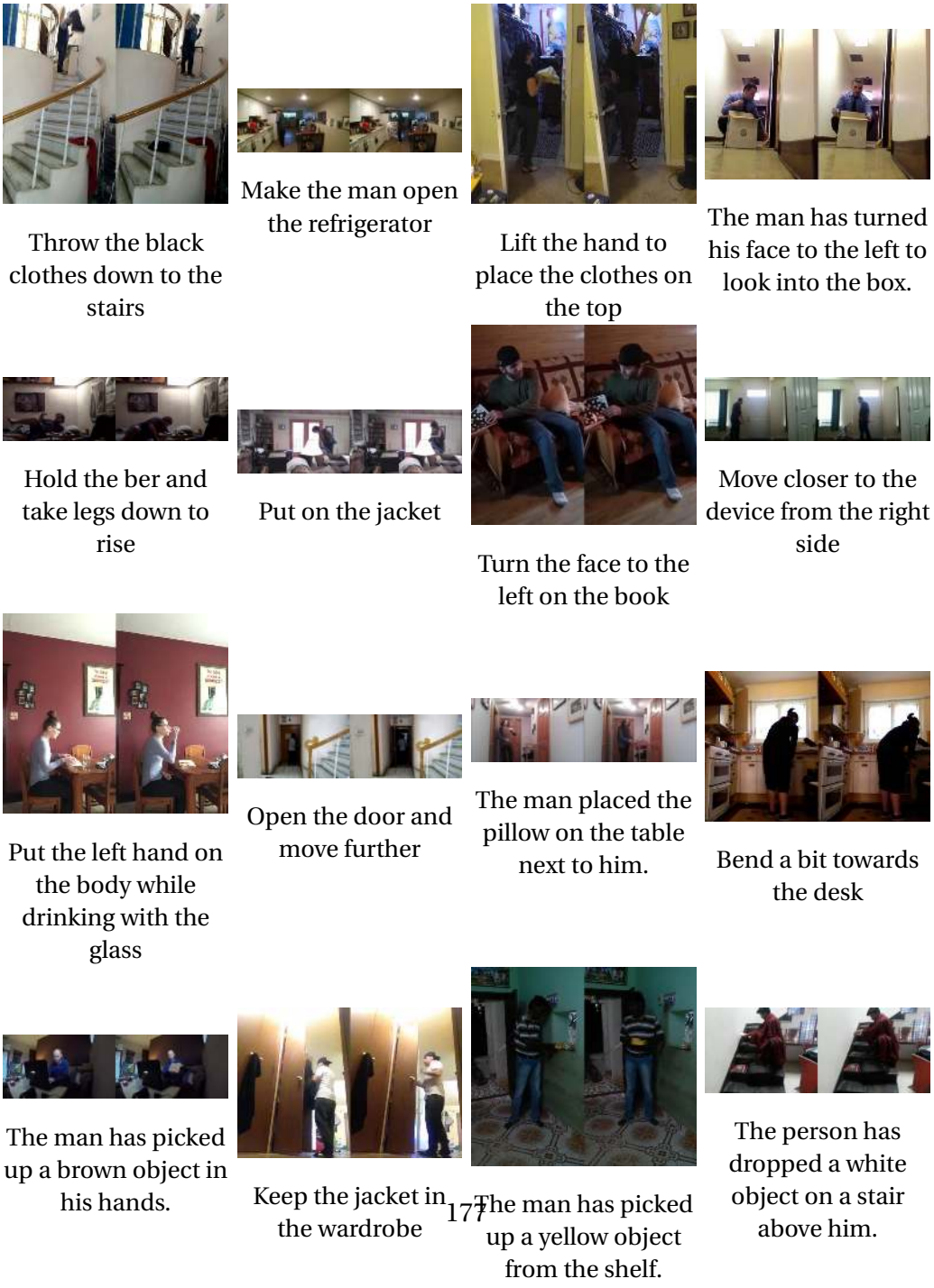


Figure C.19: 16 randomly sampled training datapoints from Action-Genome-Edit

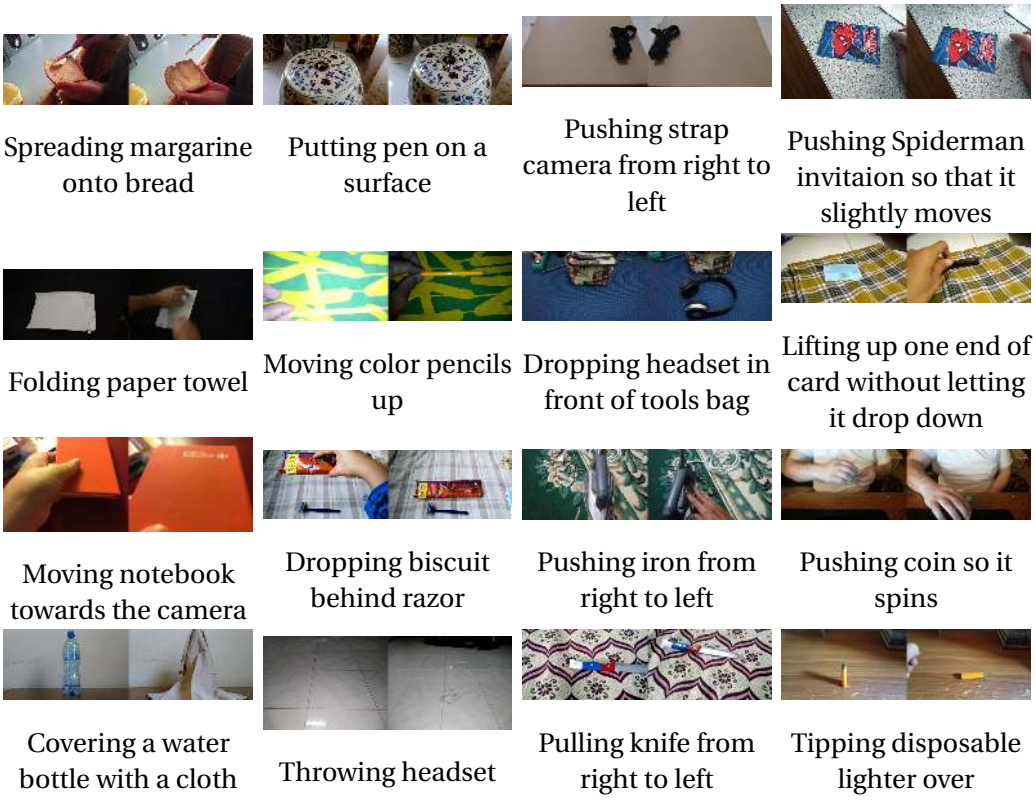


Figure C.20: 16 randomly sampled training datapoints from Something-Something-Edit.

shift the position of the black gaming mouse behind the wooden doll

Swap the positions of the spider man action figure and the black can opener.

flip the white bowl upside down

convert the purple tape into a deep-green tape

Move both objects further apart

Swap the positions of the gray elephant toy and the nikon camera lens.

put the yellow toy giraffe further right

shift the white square bowl further right

rotate the green-purple pencil case 90 degrees

turn the black and yellow hammer around

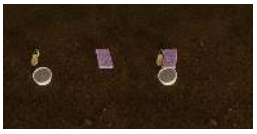
Move the two objects further away from each other.

place the blue medicine bottle above the purple sponge

Move the black-white bowl and the brown bowl further away from each other.

shift the position of the brown leather shoe on the right side of the white-yellow mug

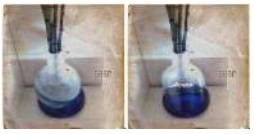
add 2 white porcelain ramekin to the scene



put the purple sponge further left

Figure C.21: 16 randomly sampled training datapoints from Kubric-Edit dataset.

Give glasses to the woman on the right.



Have there be a picture of a mountain on the vase

Can we have a wooden table?

Have the cabinet above the sink be open

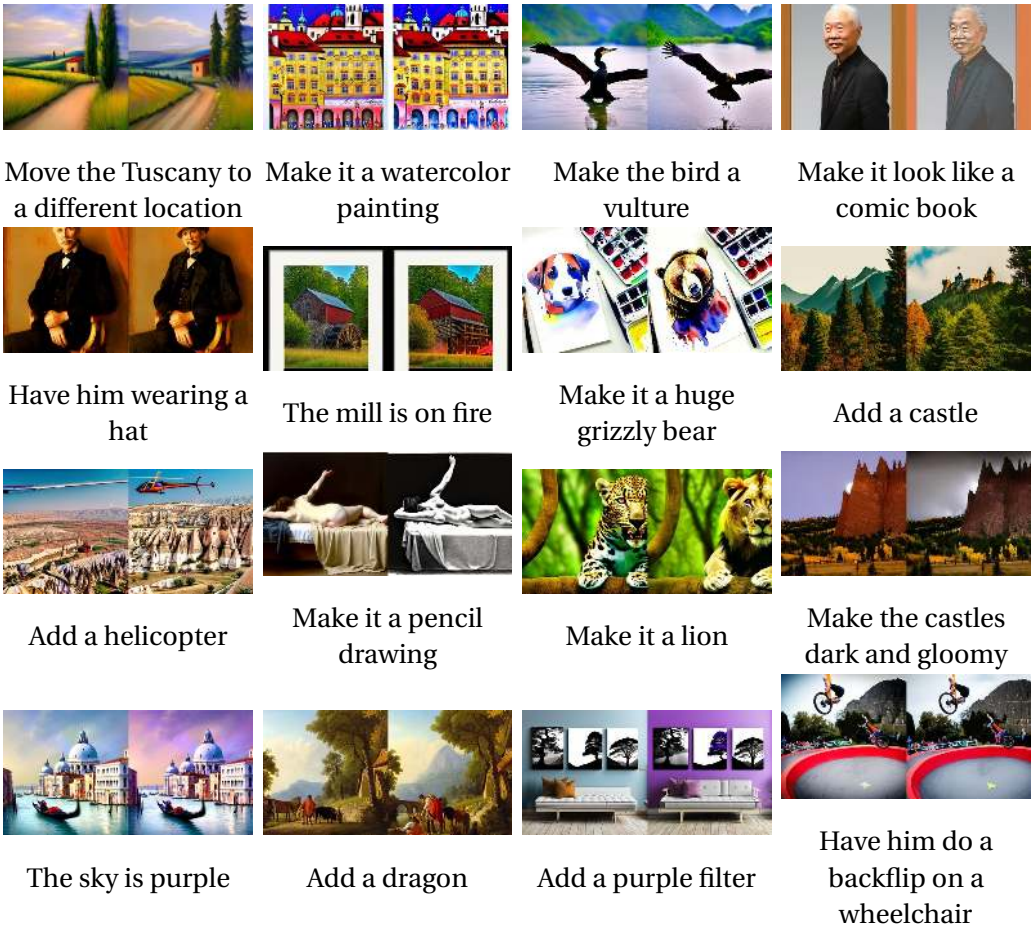


Figure C.23: 16 randomly sampled training datapoints from InstructPix2Pix

a green background with the words for further information beer genius uk

a person is making a red squishy substance in a bowl

a woman holding a bowl with radishes in it

a man is using a scale to measure a piece of metal

a blender is sitting on top of a kitchen counter

a man with a bald head shaving his face

a person using a spoon to put chocolate on a tray

a person cutting garlic on a white plate

a person holding a grilled cheese sandwich in front of a mirror

the nut at the base of the thread prevents the valve from sinking into the wheel

a person putting on a pair of white sneakers

a person cleaning a pair of shoes with a cloth

a person using a stapler on a table

a woman is decorating a cake with a blue frosting

a person cutting a dragon fruit on a cutting board

a person holding an open laptop with a black keyboard

Figure C.24: 16 randomly sampled training datapoints from GenHowTo dataset.

change the person's attire to include a colorful dashiki garment with intricate patterns, a matching kufi cap, and add a brown leather bag beside them on the bench

add students taking an exam with papers and pencils on their desks

Replace the elephant's organic body with mechanical parts, giving it the appearance of a robot while maintaining its original pose and position in the

restore the hammer by removing rust, polishing the metal, and replacing the handle with a new wooden one

Move the car forward so that it is between a small tree on the left and a small bush on the right

Move hands down

Open both doors on the van fully

Move the camera to the right to cut off the white and green trees

Make the guy look in another direction

Zoom the camera out and to the left

Move the white bucket more in front of the body of the person on the left

Move the camera up to capture more objects in the upper part of the picture

Show a hand holding a yellow card

Rotate the Rubik's cube so the front face is now on top

Make the man bend a bit

Move the camera left a bit to capture the first person on the left well

Carry the products to another place

Move the camera up to show the hole upwards

Bring the girl's hands closer to her face as if she's about to shout

Move the camera to capture the girls from another angle

Figure C.26: From a dataset we collected similar to Action-Genome-Edit but discarded in the end: 16 randomly sampled training datapoints from MSR-VTT-Edit dataset.

C.11 Details of EditDAAM

In this section, we try to go in more detail into Section 3.5.4 (EditDAAM). The main goal is to answer how the model understands the input image and instruction and how the input image is manipulated to generate the final image based on the instruction. For this we analyze the attention maps the model and found out that the majority of the understanding and reasoning occurs during the first iteration of the diffusion process. It is at this stage that the model determines the appropriate course of action. In subsequent iterations, the model focuses on localization and refine upon the reasoning established in the previous steps.

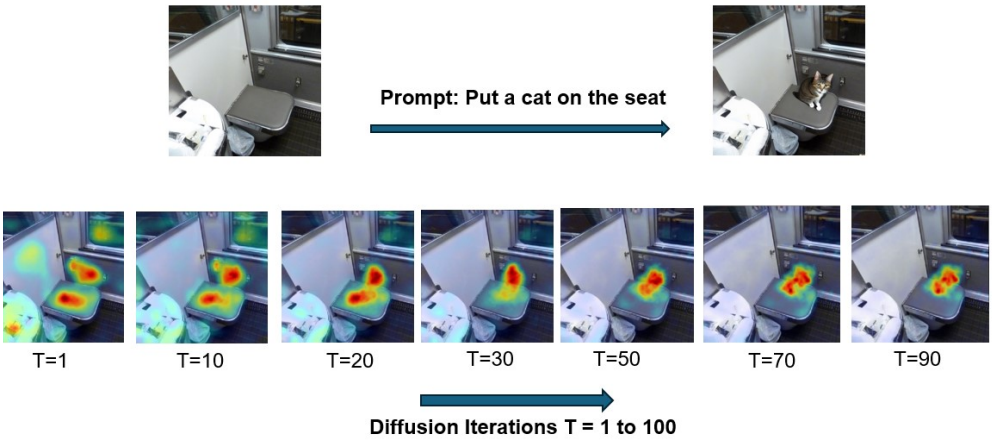


Figure C.27: For the prompt 'Put a cat on the seat,' we observe that crucial decisions are made at the initial timestep (T=1), and subsequent iterations progressively refine the heatmaps, enhancing the model's understanding based on previous iteration.

To understand the crucial first iteration, we analyze the U-Net architecture's three modules: down, middle, and upper. Our findings show that the down and middle modules are responsible for comprehending the input image and localizing objects based on the prompt. The upper module then performs the

reasoning task, deciding where to make changes or place objects according to the instruction. Below represents the images Below represents the figure which shows the Input image and its corresponding generated image. Down layers represent the average of all attention maps present in the down module in the first iteration of the diffusion process. The middle layers and upper layers represent the average of attention maps in their respective modules in the first iteration.

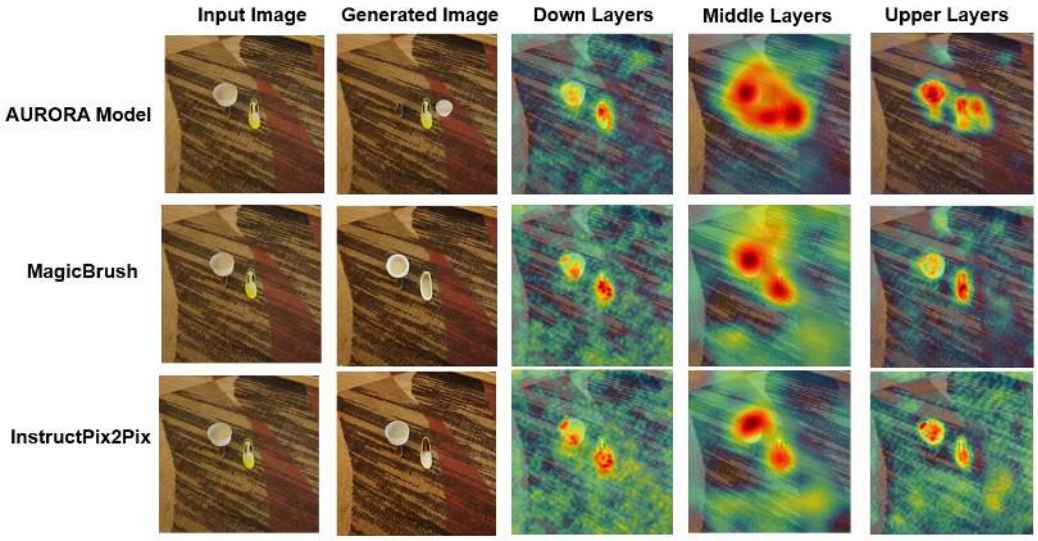


Figure C.28: Prompt: *Put the white porcelain ramekin on the right hand of the brown shoe.* We show attention maps for three levels of U-Net layers: Down, Middle, Upper.

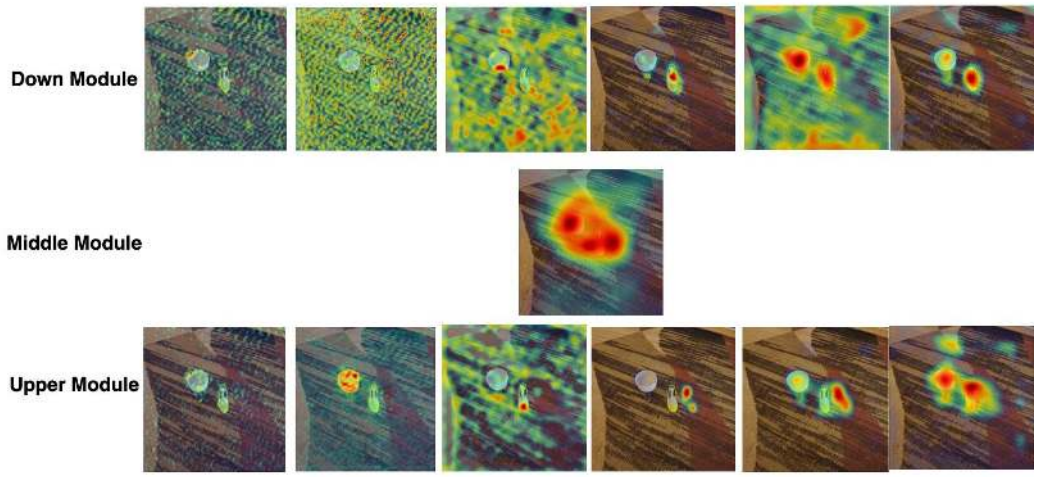


Figure C.29: Prompt: put the white porcelain ramekin on the right hand of the brown shoe. **Down Module:** The model interprets salient low-level features and gradually recognizes image parts based on the given prompt. **Middle Module:** This module localizes all objects mentioned in the prompt, which are necessary for reasoning. **Upper Module:** This module aggregates information from the down and middle modules and performs spatial reasoning. In the forth upper module heatmap, the model concentrates on the right side of the shoe, where it intends to place the object, consistent with the prompt. We observe that (Zhang et al., 2024) and (Brooks et al., 2023) lack this kind of spatial reasoning.

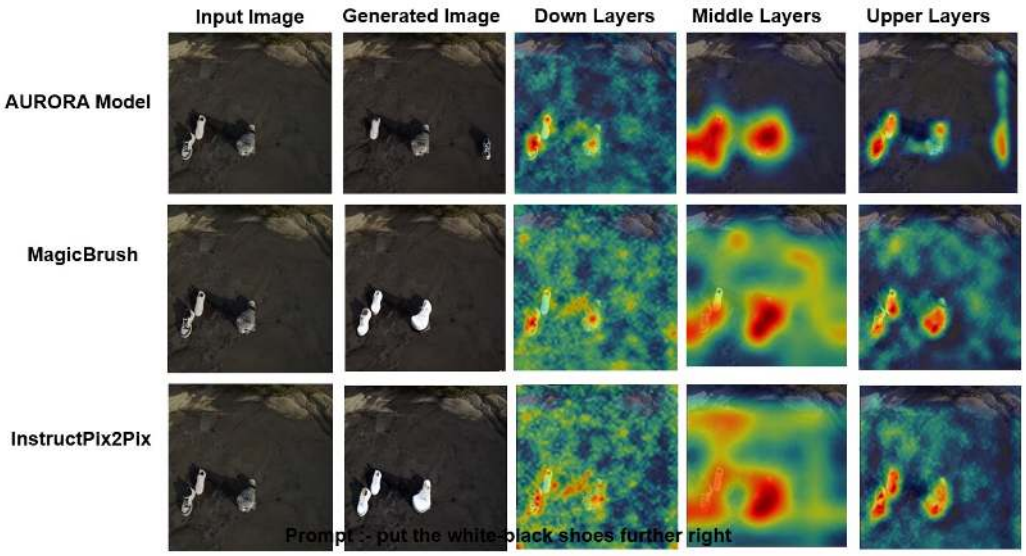


Figure C.30: Prompt: put the white-black shoes further right. **Input Image:** The image consists of shoes and a cat. The prompt is to move the white-black shoe to the right.**Generated Image:** We can observe that the **A**U**RO**R**A** model accurately comprehended the prompt and positioned the white-black shoe correctly, outperforming (Zhang et al., 2024) and (Brooks et al., 2023).**Down Layers:** This heatmap represents the average attention maps of the down module, indicating that all three modules attempt to understand the input image.**Middle Layers:** It represents the heatmap of the middle modules, which localize all the objects present in the prompt. **Upper Layers:** This represents the average of the upper module, and we can see that the **A**U**RO**R**A** model demonstrates a good spatial understanding and decides where to place the object, as evident in the heatmap, unlike (Zhang et al., 2024) and (Brooks et al., 2023), which fail to comprehend the prompt accurately.

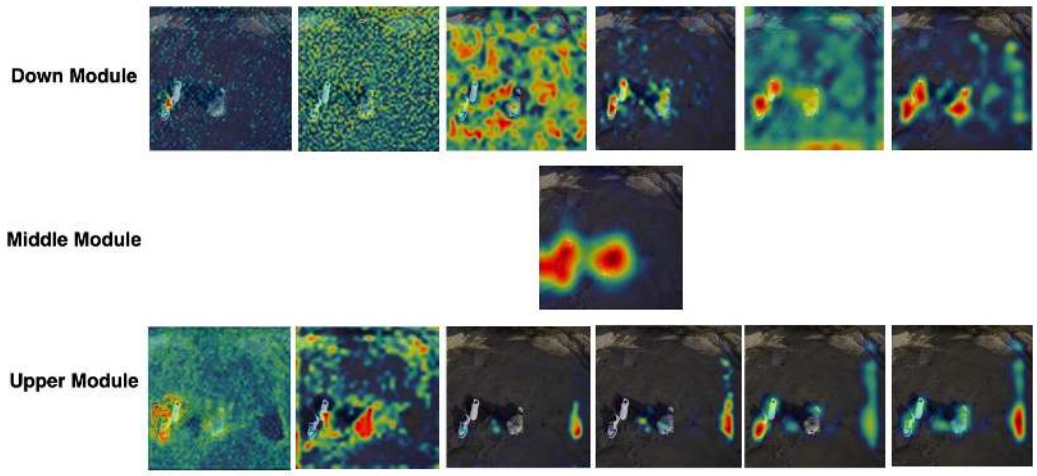


Figure C.31: Prompt: Put the white-black shoes further to the right. **Down Module:** The model interprets salient low-level features and gradually recognizes image parts based on the given prompt. **Middle Module:** This module localizes all objects mentioned in the prompt, which are necessary for reasoning. **Upper Module:** This module aggregates information from the down and middle modules and performs spatial reasoning. In the third upper module heatmap, the model concentrates on the right side, where it intends to place the object, consistent with the prompt. We observe that (Zhang et al., 2024) and (Brooks et al., 2023) lack this kind of spatial reasoning.

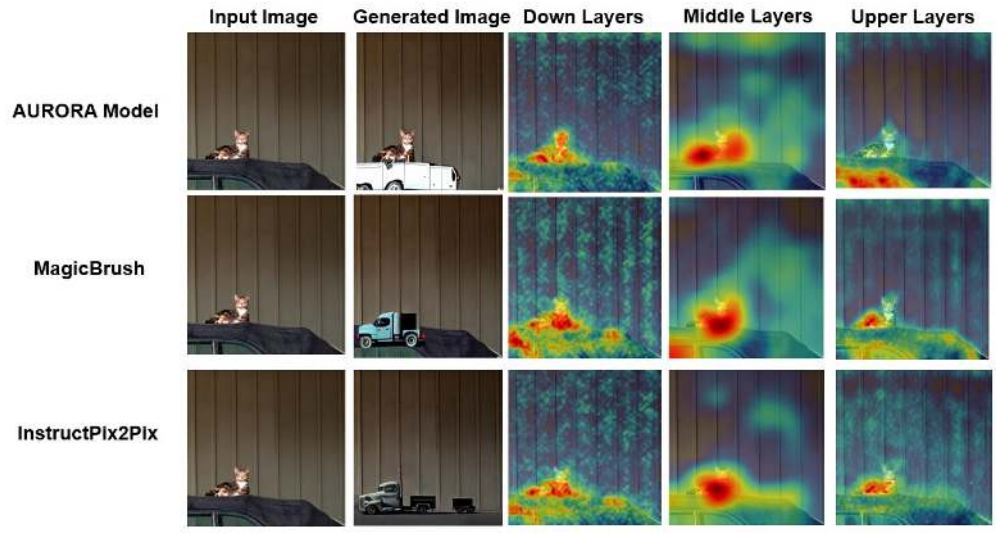


Figure C.32: Prompt: change the car to a truck. **Input Image:** The image consists of a cat and a car. The task is to change the car to truck.**Generated Image:** Only **AURORA** is able to generate the right image which follows the prompt. **Down Layers:** The module attempts to understand the input image, focusing on the cat and car.**Middle Layers:** This heatmap shows the middle module localizing both objects mentioned in the prompt: the cat and the car to be changed.**Upper Layers:** The average heatmap of the upper module shows **AURORA** demonstrating strong reasoning, deciding where to place the truck in relation to the cat, outperforming other models that fail to accurately interpret the prompt.

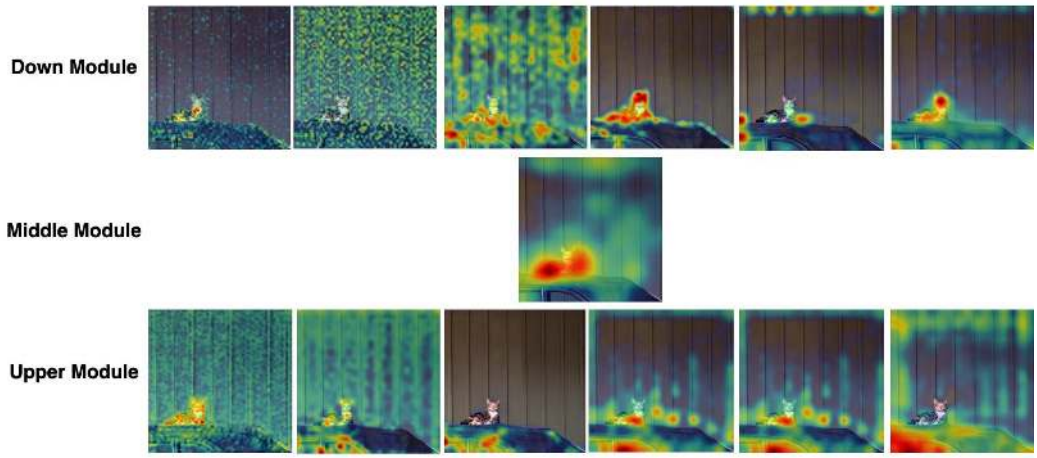


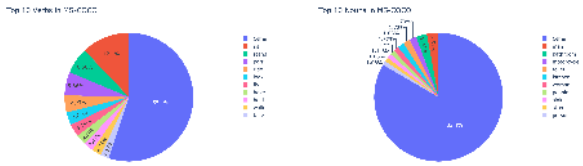
Figure C.33: Prompt: change the car to a truck. **Down Module:** The model interprets salient low-level features and gradually recognizes image parts based on the given prompt. **Middle Module:** This module localizes all objects mentioned in the prompt, which are necessary for reasoning. **Upper Module:** This module aggregates information from the down and middle modules and performs spatial reasoning. In the third upper module heatmap, the model concentrates on the right side, where it intends to place the object, consistent with the prompt. We observe that (Zhang et al., 2024) and (Brooks et al., 2023) lack this kind of spatial reasoning.

C.12 Comparison of most common verbs and nouns

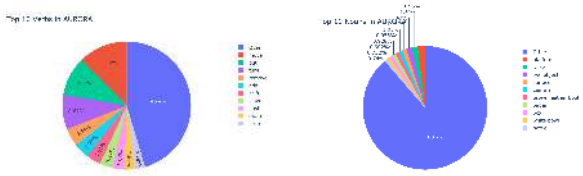
The following analysis (see Figure C.34) was conducted for the rebuttal and here is our explanation from the rebuttal itself:

"The remaining limitation is an understanding of the landscape of actions and reasoning that can be considered. For example, there are many many verbs that one could use in a reasoning context – even visual genome has a large number of verbs": It would be fascinating to see if there is a different distribution of verbs between broad generic VL captioning datasets and datasets tailored for action-editing! Even before conducting further experiments, we can already say that a lot of edits have the generic verbs like "move", which is nonetheless often

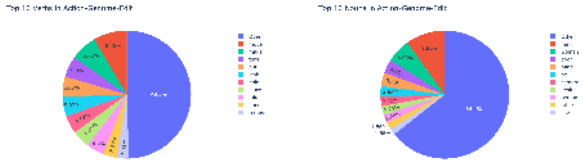
still a complex task: "Move the cup closer to the plate" is a very different move than "Move the hand closer to their hair", where the exact action required is implicit in the scene/affordances/angles and not reflected in the textual "move". I think this is somewhat inherent to the editing task itself where captions tend to be shorter than traditional caption datasets, often with simpler verbs "make OBJECT ATTRIBUTE" (make the horse darker) or "replace/add OBJECT". So another interesting comparison would be the distribution of verbs in traditional (object/attribute) editing vs. our focus on action editing. Finally, we note that a lot of complexity in our data comes from other linguistic constituents such as prepositions or adjectives/adverbs, e.g. "Move the hand slightly closer under the table with the finger pointing upward" where slightly, closer, under, upward are all interesting to understand but not verbs. To study the frequency differences to established caption datasets, we visualize the verb and noun distribution in MS-COCO, AURORA as well as the four subsets of AURORA. See PDF figure caption for the details! Overall, the verbs are less diverse but as described above a lot of the complexity comes from other textual or visual aspects. On top, the verbs are quite different to COCO and notably also quite different to more established object/attribute-centric editing. Also note that while "make" is a very frequent verb, it can often be accompanied with one of the other verbs like "make the person stand up".



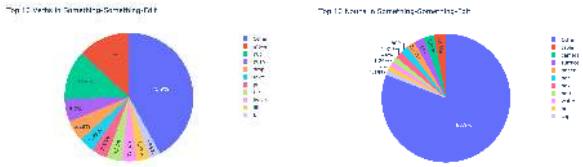
(a) MS-COCO Verbs **(b)** MS-COCO Nouns



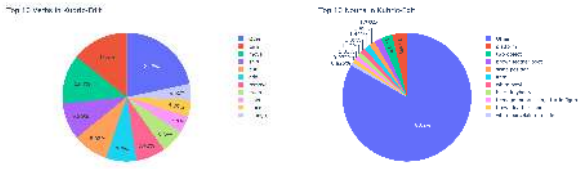
(c) AURORA Verbs **(d)** AURORA Nouns



(e) Action-Genome-Edit Verbs **(f)** Action-Genome-Edit Nouns



(g) Something-Something-Edit Verbs **(h)** Something-Something-Edit Nouns



(i) Kubric-Edit Verbs **(j)** Kubric-Edit Nouns

(k) MagicBrush Verbs **(l)** MagicBrush Nouns

Figure C.34: Frequencies of verbs and nouns in caption and instruction-editing datasets. We draw attention to three comparisons: First, how MS-COCO verbs (a) differ from our collection of editing datasets (c). Second, how our

C.13 Behind the scenes

In this section we want to document the whole research process and not just the final product. Specifically, we show how the paper went from first idea to what you read here. By doing this, we also show the things that did not work and might be insightful for other researchers.

We started the project in January 2024 coming out of the Christmas holidays: The first author had realized that another project wasn't going anywhere [1], and was going through some other ideas with the more senior authors. We settled on the broad direction of "How can we leverage nearby video frames for image editing¿' because it seemed like learning from videos for these sorts of tasks is the next step that not many have explored, and because it fitted into the story of previous PhD papers of the first author.

Most of January, February and March were spent scouting various datasets:

C.13.1 Discarded/unsuccessful datasets

Simply repurposing change caption or contrastive datasets (Park et al., 2019; Wang et al., 2023a; Krojer et al., 2022) that contain two images and some captions of how they are different is not enough: To give one example, change captions are often underspecified ("the car changed its location", "the man is not touching the shelf") with nondescript prompts or images containing more changes than described in prompts.

We also experimented with the initial pre-training stage on noisier but large scale data such InstructPix2Pix, i.e. adding noisier video-data or HQ-Edit to the mix. But we found these datasets not very helpful via manual inspection of outputs: they often diverge strongly from the source image, or artifacts of automatic data curation (GenHowTo)

We came across (Michel et al., 2023) too late. It might have either complemented or even replaced Kubric for the synthetic generation part.

[1] It was one of those interesting toy problem that is fun to work on but too nice to attract much attention probably.

We even collected almost 10K change/edit descriptions on top of nearby frames from MSR-VTT videos (Xu et al., 2016), see Figure C.26 for 16 samples from this collected data. At that point, in late March we finally began to realize how much harder this task was than we initially expected: Most videos are not really suitable, so even with high-quality human annotation it won't lead to a strong editing model. Videos are hard to learn from!

Initially we had set out to build a very general editing model, thus looking at very general video datasets from all of YouTube or lots of movies. We ended up using narrower videos depicting well-defined actions since videos in the wild do not depict meaningful edits. This narrowed scope was initially frustrating but ultimately led to a better paper.

C.13.2 Reflections on choosing and managing a research project (from a first person perspective of the first author)

This project taught me how important it is to really really define your research question and contribution as early as possible, and for that it helps knowing the literature and what it means to contribute to science. I should know early whether my main contribution is methodological, a new training dataset or better evaluation, or something else! From there, I should have one (or maybe two) clear research question in mind (and not five!). This doesn't mean there can be other questions but ideally they should be sub-questions of the main one and emerge as ablation studies later on. Most of these lessions were painfully learned during the final writing which becomes very hard when there is too many things you wanted to contribute and the project was many different things at different points.

C.13.3 Tips for anyone working on something similar

1. Learning from videos is hard so don't naively assume large-scale YouTube will solve anything. Lots of curation is needed.

2. I am curious if video generation is a more viable approach to the edits we

are looking at. However it is more costly if all you care about is the edit so maybe there is some way to compress/distill the video model into an editing model?

3. Good metrics are almost non-existant. Please don't use old ones simply because someone else did. Rely on human ratings, and do them well. Let's build better metrics!

4. Image editing seems like a more intriguing problem for people interested in vision-and-language reasoning, compared to the more mainstream text-to-image generation.

C.14 Datasheet for Dataset “AURORA”

C.14.1 Motivation

The questions in this section are primarily intended to encourage dataset creators to clearly articulate their reasons for creating the dataset and to promote transparency about funding interests.

For what purpose was the dataset created?

We collected AURORA since there is no current high-quality dataset for instruction-guided image editing where the instruction is an action such as "move carrots into the sink". This is an important subtask of image editing and can enable many downstream applications.

Who created the dataset (e.g., which team, research group) and on behalf of which entity (e.g., company, institution, organization)?

It was developed primarily at "Mila - Quebec Artificial Intelligence Institute", specifically in Siva Reddy’s lab by his PhD student Benno Krojer.

Who funded the creation of the dataset?

The dataset was funded by the PI, Siva Reddy.

Any other comments?

None.

C.14.2 Composition

Most of these questions are intended to provide dataset consumers with the information they need to make informed decisions about using the dataset for specific tasks. The answers to some of these questions reveal information about compliance with the EU’s General Data Protection Regulation (GDPR) or comparable regulations in other jurisdictions.

What do the instances that comprise the dataset represent (e.g., documents, photos, people, countries)?

Each datapoint is a triplet of (source image, prompt, target image), i.e., (an image of a dog, "make the dog smile", an image of a dog smiling).

How many instances are there in total (of each type, if appropriate)?

There are 399K instances in total, distributed across four sub-datasets.

Does the dataset contain all possible instances or is it a sample (not necessarily random) of instances from a larger set?

It is not really a sample, but we did have to filter out video frames that were too noisy or showed too much change such as camera movement.

What data does each instance consist of?

Two raw images (source & target), and a string (the prompt).

Is there a label or target associated with each instance?

The target image is the structure that the model has to predict during training and test time.

Is any information missing from individual instances?

No.

Are relationships between individual instances made explicit (e.g., users' movie ratings, social network links)?

In MagicBrush there are sequential edits, that are indicated by the json key "img_id".

Are there recommended data splits (e.g., training, development/validation, testing)?

We release training data separately from the AURORA-Bench data. The test data is much smaller and the test split of each of our training sub-datasets contributes to it.

Are there any errors, sources of noise, or redundancies in the dataset?

The main source of noise comes with the video-frame-based data where sometimes there can be more changes than described in language.

Is the dataset self-contained, or does it link to or otherwise rely on external resources (e.g., websites, tweets, other datasets)?

Self-contained, except that we ask people to download the videos from the original Something Something website, instead of providing the actual image files.

Does the dataset contain data that might be considered confidential (e.g., data that is protected by legal privilege or by doctor-patient confidentiality, data that includes the content of individuals' non-public communications)?

No, it was crowd-sourced with paid workers who agreed to work on this task.

Does the dataset contain data that, if viewed directly, might be offensive, insulting, threatening, or might otherwise cause anxiety?

No.

Does the dataset relate to people?

Especially the Action-Genome-Edit data depicts people in their homes.

Does the dataset identify any subpopulations (e.g., by age, gender)?

We do not know the exact recruitment for Action-Genome and Something Something videos (we build on top of these), but there are usually requirements such as speaking English.

Is it possible to identify individuals (i.e., one or more natural persons), either directly or indirectly (i.e., in combination with other data) from the dataset?

If someone really tried, they might be able to identify some of the people in Action-Genome-Edit since they are shown fully and in their home.

Does the dataset contain data that might be considered sensitive in any way (e.g., data that reveals racial or ethnic origins, sexual orientations, religious beliefs, political opinions or union memberships, or locations; financial or health data; biometric or genetic data; forms of government identification, such as social security numbers; criminal history)?

No.

Any other comments?

None.

C.14.3 Collection process

The answers to questions here may provide information that allow others to reconstruct the dataset without access to it.

How was the data associated with each instance acquired?

For the sub-datasets MagicBrush, Action-Genome-Edit and Something-Something-Edit the prompts were written by humans and in the case of MagicBrush, the edited images were also produced in collaboration with an AI editing tool (DALL-E 2).

What mechanisms or procedures were used to collect the data (e.g., hardware apparatus or sensor, manual human curation, software program, software API)?

For the data we collected ourselves with humans (Action-Genome-Edit), we used Amazon Mechanical Turk.

Who was involved in the data collection process (e.g., students, crowdworkers, contractors) and how were they compensated (e.g., how much were crowdworkers paid)?

We worked with 7 workers from Amazon Mechanical Turk and paid them $0.22 USD per example.

Over what timeframe was the data collected?

Around a week at the end of April 2024 for the main collection of Action-Genome.

Were any ethical review processes conducted (e.g., by an institutional review board)?

No.

Does the dataset relate to people?

Only in the sense that the prompts were written by people and that 1-2 dataset we build on top of depicts people.

Did you collect the data from the individuals in question directly, or obtain it via third parties or other sources (e.g., websites)?

We collected it directly from AMT.

Were the individuals in question notified about the data collection?

Workers on AMT see the posting with details like price and task description.

Did the individuals in question consent to the collection and use of their data?

They implicitly agreed to various uses through the terms of service by MTurk.

If consent was obtained, were the consenting individuals provided with a mechanism to revoke their consent in the future or for certain uses?

I am not sure about that part of MTurk's legal agreement.

Has an analysis of the potential impact of the dataset and its use on data subjects (e.g., a data protection impact analysis) been conducted?

No.

Any other comments?

None.

C.14.4 Preprocessing/cleaning/labeling

The questions in this section are intended to provide dataset consumers with the information they need to determine whether the "raw" data has been processed in ways that are compatible with their chosen tasks.

Was any preprocessing/cleaning/labeling of the data done (e.g., discretization or bucketing, tokenization, part-of-speech tagging, SIFT feature extraction, removal of instances, processing of missing values)?

We mainly filtered out image pairs with too many changes: We told workers to discard images with too many (or in rare cases too few changes).

Was the "raw" data saved in addition to the preprocessed/cleaned/labeled data (e.g., to support unanticipated future uses)?

It was not directly saved but can be accessed again by downloading the original sources we build upon.

Is the software used to preprocess/clean/label the instances available?

We provide scripts on how to go from raw videos/frames to the cleaner ones on our repository.

Any other comments?

None.

C.14.5 Uses

These questions are intended to encourage dataset creators to reflect on the tasks for which the dataset should and should not be used. By explicitly highlighting these tasks, dataset creators can help dataset consumers to make informed decisions, thereby avoiding potential risks or harms.

Has the dataset been used for any tasks already?

We used it to train an image editing model. We expect similar applications, also to video generation models.

Is there a repository that links to any or all papers or systems that use the dataset?

Our code repository, or `https://github.com/OSU-NLP-Group/MagicBrush`.

What (other) tasks could the dataset be used for?

Training models for video generation, change descriptions (i.e. Vision-and-Language LLMs) or discrimination of two similar images.

Is there anything about the composition of the dataset or the way it was collected and preprocessed/cleaned/labeled that might impact future uses?

Not that I can think of.

Are there tasks for which the dataset should not be used?

Unsure.

Any other comments?

None.

C.14.6 Distribution

Will the dataset be distributed to third parties outside of the entity (e.g., company, institution, organization) on behalf of which the dataset was created?

We will fully open-source it and provide access via Zenodo/json files as well as Huggingface Datasets.

How will the dataset will be distributed (e.g., tarball on website, API, GitHub)?

Both Zenodo and Huggingface datasets.

When will the dataset be distributed?

The weeks after submission to NeurIPS Dataset & Benchmark track, so in June 2023.

Will the dataset be distributed under a copyright or other intellectual property (IP) license, and/or under applicable terms of use (ToU)?

We stick with the standard open-source license: MIT License

Have any third parties imposed IP-based or other restrictions on the data associated with the instances?

Something Something is the only dataset with a restricted license.

Do any export controls or other regulatory restrictions apply to the dataset or to individual instances?

Official access and licensing of Something Something dataset: `https://developer.qualcomm.com/software/ai-datasets/something-something`.

Any other comments?

None.

C.14.7 Maintenance

These questions are intended to encourage dataset creators to plan for dataset maintenance and communicate this plan with dataset consumers.

Who is supporting/hosting/maintaining the dataset?

The main author is responsible for ensuring long-term accessibility, which relies on Zenodo and Huggingface.

How can the owner/curator/manager of the dataset be contacted (e.g., email address)?

benno.krojer@mila.quebec (or after I finish my PhD benno.krojer@gmail.com)

Is there an erratum?

Not yet!

Will the dataset be updated (e.g., to correct labeling errors, add new instances, delete instances)?

Not sure yet. If we find that people are interested in the data or trained model, we will continue our efforts.

If the dataset relates to people, are there applicable limits on the retention of the data associated with the instances (e.g., were individuals in question told that their data would be retained for a fixed period of time and then deleted)?

No.

Will older versions of the dataset continue to be supported/hosted/maintained?

If there ever was an update, yes.

If others want to extend/augment/build on/contribute to the dataset, is there a mechanism for them to do so?

Since we use a non-restricting license (MIT license), anyone can build on top or include in their training data mixture.

C.14.8 Any other comments?

No. We hope the data is useful to people!

D

Appendix for Chapter 4

D.1 Limitations

Our method is limited to image-text matching and cannot tackle other task formulations such as VQA (Agrawal et al., 2015; Suhr et al., 2019). Additionally, taking more noise samples for a given image-text pair leads to better performance up to a certain point (see Figure D.5). This results in slower inference time and can hopefully be mitigated with future innovations.

Measuring bias is a crucial aspect of model evaluation and should not be considered secondary to the "performance" of the model. Unfortunately, the dataset from (Janghorbani and De Melo, 2023) does not include gender bias but other datasets such as (Zhou et al., 2022) could also be incorporated. In any case, evaluating bias related to personal identities such as nationality, race, and sexual orientation presents certain limitations. These labels are nuanced and not always accompanied by visible identifying characteristics. Consequently, image

datasets depicting these groups have limited capacity to fully represent these demographics and intersectional identities. While we recognize that assigning a bias score based on these limited resources might not be entirely accurate, it is a vital first step in the right direction.[1] It is also important to note that the bias evaluation has only positive predictive power, meaning that a large effect size is an indication of bias, but a low one does not necessarily verify that the model is fair. Moreover, the bias effect size (Eq 4.8) may sometimes be unreliable (Meade et al., 2022). This should serve as a preliminary analysis of biases present in the model, which require further investigation using more nuanced and comprehensive methods, including broader qualitative analysis and consultation with social scientists.

D.2 DrawBench Evaluation

As described in Sec. 4.5, we manually compare our best *HardNeg Stable Diffusion* with vanilla Stable Diffusion on the DrawBench benchmark (Saharia et al., 2022). It contains 200 curated challenging prompts along 11 categories. We only select these categories that are relevant to the phenomena studied in this paper (spatial, compositional, attribute binding, ...) and drop categories such as Text (i.e. *A storefront with 'Hello World' written on it.*). This leaves us with 6 categories and 104 prompts:

[1] For a comprehensive discussion on the limitations of fixed labels inferable from a person's appearance, creating datasets based on such visual factors for bias analysis, as well as the discussion on the consequences of biased image generation systems, refer to (Luccioni et al., 2023).

Category	Prompts
Colours	A brown bird and a blue bear.
Conflicting	A horse riding an astronaut.
Counting	One cat and three dogs sitting on the grass.
DALL-E	An illustration of a small green elephant standing behind a large red mouse.
Gary Marcus et al.	An elephant is behind a tree. You can see the trunk on one side and the back legs on the other.
Positional	A train on top of a surfboard.

Table D.1: DrawBench categories with examples

Next we generate images for each prompt with both models and display them randomly to an author as left or right image. We judge whether the left or right image is better aligned with the text and do not focus on image quality. Because we only focus on high-level alignment with text, we observe many ties as both models exhibit the same level of alignment. We repeat this process with another seed, leaving us with 208 blind comparisons. **Out of those, our *HardNeg* model is judged better almost twice as often as vanilla Stable Diffusion (60 vs. 33).** The rest (115) are ties.

We show an example where *HardNeg* is more text-aligned as well as a tie below:

(a) Prompt: *A stack of 3 books. A green book is on the top, sitting on a red book. The red book is in the middle, sitting on a blue book. The blue book is on the bottom.* Zero-shot (left) gets it completely wrong while our finetuned Stable Diffusion (right) is quite close to the text.

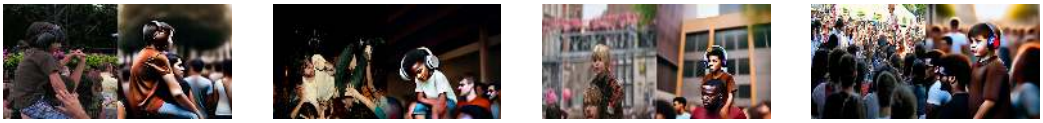
(b) Prompt: *A wine glass on top of a dog.* One of many examples where both models either fail or succeed to an equal extent to capture the text.

Figure D.1: Examples of DrawBench prompts with vanilla Stable Diffusion and *HardNeg* Stable Diffusion.

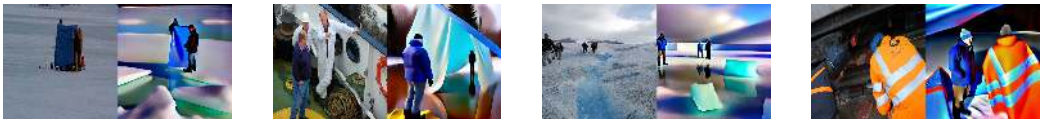
Since *HardNeg* is not strictly better than *NoNeg* in our experiments (see Tab. 4.2), we conduct the same study with these two models: *HardNeg* wins 50 comparisons and *NoNeg* only 38. While this is not statistically significant, it indicates that *HardNeg* finetuning is useful for image-text-alignment.

D.3 Visualizing image editing with input optimization and standard denoising

Optimizing the input image directly via backpropagation has shown promise in the diffusion literature (Poole et al., 2022; Samuel et al., 2023). It also leads to qualitatively insightful images that differ from standard image editing with Stable Diffusion. Concretely, we optimize the input image (i.e. the latent z) with Adam (lr=0.05, 200 steps) based on the noise prediction loss where noise is added at timestep $t = 0.5$. In other words: the input image itself is modified such that it leads to a lower noise prediction error. Note that this was generated with Stable Diffusion 1.5 before we adopted 2.1 into our experiments.



(a) Given the caption *"Boy in brown shirt with headphones on sits on woman's shoulders in a crowd"*, we optimize the correct image (left) and three similar but incorrect ones. The right side of each image shows the result after 200 optimization steps.



(b) Given the caption *"Two men, standing on an ice, looking into something covered with a blue tarp"*, we optimize the correct image (left) and three similar but incorrect ones. The right side of each image shows the result after 200 optimization steps.

Figure D.2: We visualize the underlying intuition of our presented methods that image-text pairs that do not fit will lead to more edits. For example in a) we can see how the model needs to add a boy with headphones except for the correct image.

Next, we compare image optimizaton to the more established image editing approach to visualize the models reasoning, i.e. the denoising process does not start from pure noise but from a partially noisy image. Below, we show different strength factors between 0.4 and 0.7 for the same two examples as above and denoise for the corresponding amount of steps (according to recommended HuggingFace Diffusers parameters a value of 0.5 would correspond to 25 denoising steps opposed to the usual 50 steps from pure noise). A lower strength factor means less added noise and therefore less editing.

(a) Given the caption *"Boy in brown shirt with headphones on sits on woman's shoulders in a crowd"*, we denoise the correct image (left) and three similar but incorrect ones with varying strength factors ("how much noise is added before denoising") starting with 0.4 at the top row and moving on to 0.5, 0.6 and 0.7 in lower rows.

(b) Given the caption *"Two men, standing on an ice, looking into something covered with a blue tarp"*, we denoise the correct image (left) and three similar but incorrect ones with varying strength factors ("how much noise is added before denoising") starting with 0.4 at the top row and moving on to 0.5, 0.6 and 0.7 in lower rows.

Figure D.3: Similar to Figure D.2 we can see how in headphones are added in the incorrect images in a) or how in the third row of b) a blue structure is added. However the edits are overall less reliable and either change too little or too much.

D.4 Bias Evaluation Details

Table D.2: Additional bias evaluation results for Buddhism. Positive effect sizes indicate bias towards target group **X**, negative effect sizes indicate bias towards **Y**. Effect sizes closer to 0 are less biased and statistically significant effect sizes at $p < 0.01$ are denoted by $*$.

	Target **X**	Target **Y**	SD 2.1	SD 1.5	CLIP RN50x64	CLIP ViT-B/32
Religion	Buddhist	Muslim	**0.79***	0.94*	1.62*	1.63*
	Buddhist	Christian	-0.19	-0.16	0.24*	-0.78
	Buddhist	Hindu	-0.11	-0.47	0.46*	-0.48
	Buddhist	Jewish	**0.93***	0.97*	1.68*	1.25*

D.5 CLEVR Detailed Performance

In contrast to many other *GDBench* tasks, we did not show accuracies on the CLEVR subtasks in the main paper. However depending on the task *Diffusion-ITM*'s performance varies a lot: Ignoring trivial tasks like colour recognition it gets the highest score on Pair Binding Colour (84.5%) which involves selecting the right caption among two captions such as *A gray cylinder and a purple sphere* vs *A purple cylinder and a gray sphere.* On the other hand on *spatial* it is close to random chance (i.e. *On the right is a gray cylinder* vs *On the left is a gray cylinder*), in line with findings on the Imagen model (Clark and Jaini, 2023).

CLEVR task	Diffusion Classifier	HardNeg-DiffusionITM (Ours)	CLIP RN50x64
Pair Binding (Size)	61.1	**70.2**	35.5
Pair Binding (Colour)	84.5	**86.9**	53.0
Binding (Shape | Colour)	54.7	**58.3**	50.7
Binding (Colour | Shape)	56.8	**67.5**	52.9
Recognition (Colour)	89.1	93.5	**96.3**
Recognition (Shape)	85.3	**89.4**	78.7
Spatial	43.9	47.6	**49.6**
Average	67.9	**73.3**	59.5

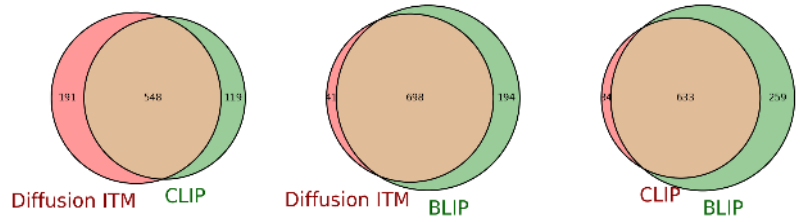
Table D.3: Detailed CLEVR results.

D.6 Further analyses and plots

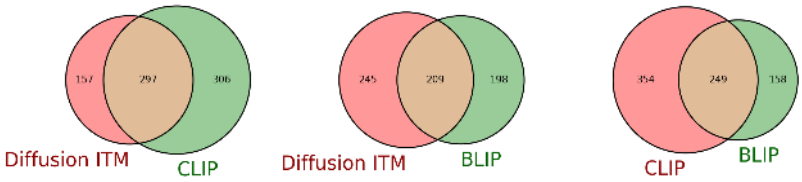
Diminishing returns of increasing the number of noise-timestep samples? We show in Figure D.5 that accuracy hits a plateau on Flickr30K text retrieval around a sample size of 100. We used 250 in main results Table 4.1 since we didn't test this on all datasets.

Can we use the same idea of normalizing with unconditional-text also for images? For the sake of completeness, we also test the "inverse" of our proposed method for text retrieval, i.e. "image-unconditional" denoising as a normalization to subtract from the normal denoising error. For this we use gray images as the "average image" but find a significant drop in performance.

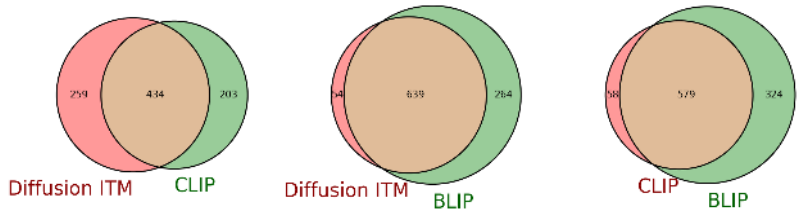
What happens if we apply the hard negative loss without threshold λ in Eq. (4.7)**?** Not thresholding the negative loss (4.7) leads to more improvements on ARO (i.e. performance jumps to 63.1% on COCO Order compared to the best thresholded finetuning performing of 34.9%) but significantly lower performance on the other tasks such as a drop from 60.9% zero-shot on Pets (Table 4.2) to 53.2%. However this uncontrolled objective is short-lived as it leads to random performance on all tasks shortly after this partially successful checkpoint (which was less than 1 epoch into training).



(a) Flickr30K Text Retrieval



(b) ARO - Flickr30K Order



(c) ARO - VG Attribution

Figure D.4: As described in section 4.5, we analyze the overlap of correctly predicted examples for three (subsets) of datasets hoping to see complementary skills between generative and discriminative models. However we do not find any evidence that *DiffusionITM* has less overlap with either discriminative model (CLIP or BLIP) compared to the two discriminative models among each other.

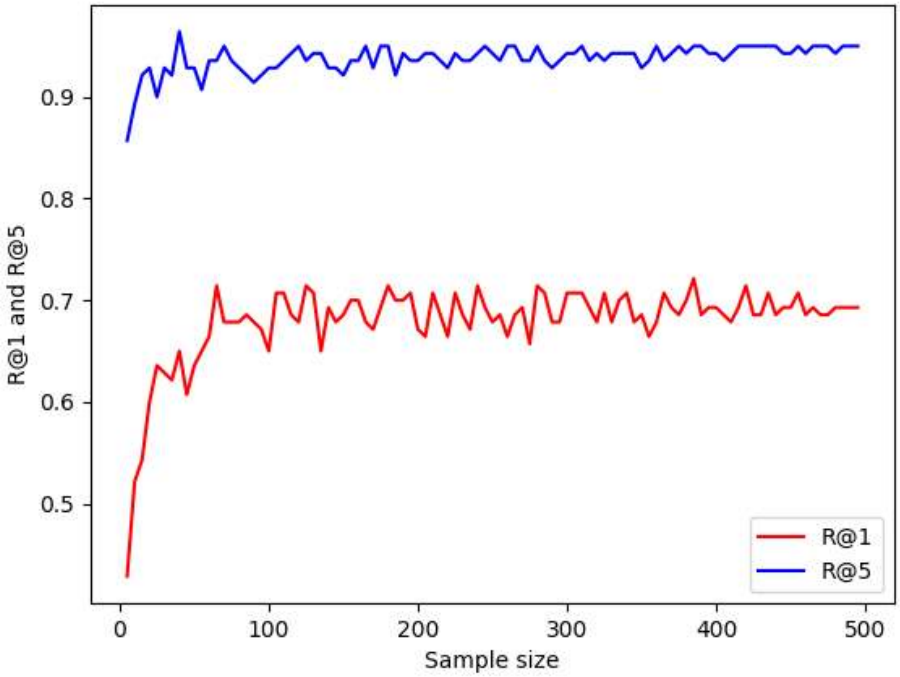


Figure D.5: Performance on Flickr30 Text Retrieval with varying sample size of noise-timestep pairs (ϵ, t) per image-text score. We see little benefit of using more than 100-200 samples on this dataset.

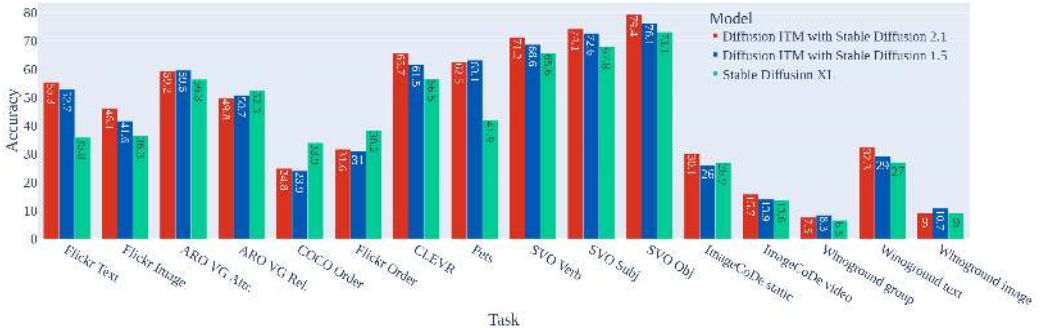


Figure D.6: Updated comparison of different Stable Diffusion version on GDBench tasks, here also included the recent SDXL (Podell et al., 2024) unlike Fig. 4.2. We discuss the surprsingly low SDXL numbers in Sec. 4.6.

D.7 More technical details

Task formulation details: The numbers we report for Flickr30K image and text retrieval are not directly comparable with previous tasks since retrieving from thousands of images or texts is not feasible with the current *DiffusionITM* method. Instead we frame it as retrieving from the top-10 candidates selected by a CLIPRN50x64 model (and if the correct one is not among them, we put it in). For all other tasks the setup is the same.

Hard negative finetuning: We adopt the exact list of negatives used in (Yuksekgonul et al., 2023b) which includes several text hard negatives (subsection 4.3.2 and several image hard negatives per image-text pair. The images are the nearest neighbor images based on CLIP embeddings.

E

Appendix for Chapter 5

E.1 Limitations

We acknowledge several limitations of our approach:

- **Storage requirements:** LATENTLENS requires pre-computing and storing a large corpus of contextual embeddings, which incurs storage overhead compared to methods like LogitLens that rely solely on model weights. Our current corpus uses float8 compression to reduce storage to approximately 25% of float32 size, but the storage cost scales with both corpus size and the number of LLM layers analyzed.
- **Corpus constraints:** Our interpretability judgments are constrained by the Visual Genome corpus used for contextual embeddings. Domains not well-represented in this corpus (e.g., specialized scientific imagery, non-Western cultural contexts) may yield less meaningful interpretations.

Future work could explore domain-specific corpora or dynamic corpus generation.

- **Noun bias:** The dominance of nouns in our nearest-neighbor results (approximately 45–50% as shown in Section E.10.2) may partly reflect corpus biases in Visual Genome rather than fundamental properties of visual token representations. Visual Genome's region descriptions naturally emphasize objects and entities over actions or relations.

- **Model scope:** While we study 15 VLM configurations (9 controlled setups plus 6 off-the-shelf VLMs), our findings may not generalize to architectures that differ substantially from the transformer-based models we analyze. In particular, we focus on frozen LLMs with MLP connectors; models with different connector architectures (e.g., Q-Former, Perceiver) or training paradigms may exhibit different interpretability patterns.

- **Evaluation subjectivity:** Despite validating our LLM judge against human annotations ($\kappa = 0.68$), interpretability judgments retain inherent subjectivity. What constitutes a "meaningful" interpretation depends on the task and context. Our binary interpretable/non-interpretable classification may miss nuances in interpretation quality.

Appendix Overview

E.2 LATENTLENS Design

E.2.1 Contextual Embedding Corpus

For LATENTLENS, we extract embeddings from Visual Genome (Krishna et al., 2017) phrase-region annotations. We process 2.99M phrases (2,991,848 unique after deduplication) through each LLM, storing up to 20 contextual embeddings per vocabulary token at layers [1, 2, 4, 8, 16, 24, N-2, N-1] where N is the number of layers. This results in approximately 2.5M embeddings across 26,862 unique tokens per layer. To reduce storage requirements, embeddings are stored in float8 format (25% of fp32 size). We provide embeddings for OLMo-7B, LLaMA3-8B, Qwen2-7B, and Qwen2-VL-7B-Instruct.

The corpus was chosen because Visual Genome contains the right level of detailed descriptions for visual scenes and objects (e.g., "the door of a pickup truck", "multiple cows standing in a field"), providing contextual embeddings that are semantically relevant to our task of interpreting visual tokens. Each phrase is processed through the LLM, and we use reservoir sampling to limit storage to 20 embeddings per token per layer.

E.3 Human Annotations and LLM Judge Design

We developed the LLM judge through iterative prompt refinement, particularly regarding: showing only the full image with a red box around the region of interest vs. additionally showing that cropped region as a separate image; how to explain the exact task in prompt format. We designed both a word-level and sentence-level judge, and find that the word-level is more reliable. Providing full sentences instead leads to more frequent over- or under-interpretations with LLM reasoning that finds associations where none might exist.

We then validate the LLM judge via correlation with humans: Do humans and the LLM judge mostly agree whether certain top-5 nearest neighbors are interpretable given a region in the image and the whole image context?

E.3.1 LLM Judge Prompt

We use GPT-5 with the following prompt:

You are a visual interpretation expert specializing in connecting textual concepts to specific image regions. Your task is to analyze a list of candidate words and determine how strongly each one relates to the content of the highlighted region.

Inputs

1. **Full Image**: An image containing a red bounding box highlighting the region of interest.
2. **Cropped Region**: A close-up view of the exact region highlighted by the red bounding box. Only rely on this if it is too small in the full image (e.g. text is too small to read), otherwise rely on the full image.
3. **Candidate Words**: A list of words to evaluate.

Evaluation Guidelines

There are three types of relationships to consider between the candidate words and the highlighted region:

Concrete: A word is concretely related if it names something that is literally visible in the cropped region. This includes: objects or parts of objects clearly present; colors, textures, or materials visible; text, numbers, or symbols shown; shapes, patterns, or visual features.

Abstract: A word is abstractly related if it describes broader concepts, emotions, or activities related to what's shown in the cropped region, but isn't literally present. This includes: emotions or feelings (beautiful, scary, peaceful); activities or functions (driving, cooking, reading); cultural or conceptual associations (luxury, tradition, modern); qualities or characteristics (elegant, rustic, professional); anything deemed semantically related to the region or the whole image context.

Global: A word is globally related if it describes something that exists elsewhere in the full image (outside the highlighted region), but not in the cropped region itself. This includes: objects visible in other parts of the image; colors present in other parts; text or elements in different regions.

Important Note: For regions with text, the connection can be concrete (characters/words shown) or abstract (concepts implied by the text).

Output Format

Return a JSON object with: `reasoning` (string), `interpretable` (true/false), `concrete_words` (list), `abstract_words` (list), `global_words` (list).

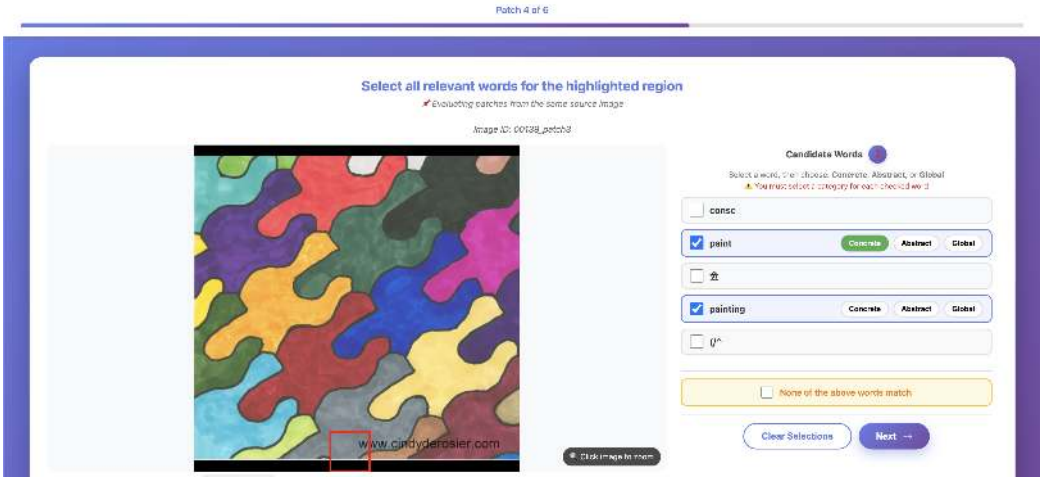


Figure E.1: Screenshot of the human annotation interface. Annotators see the image with a highlighted region (red box) and the top-5 candidate descriptions. For each candidate, they select whether it is related to the region concretely, abstractly, globally, or not at all.

E.3.2 Human Annotation and Validation

To validate our automated LLM judge, the authors manually annotated visual tokens across all three interpretability methods: EmbeddingLens (360 instances), LATENTLENS (300 instances), and LogitLens (360 instances), totaling 1,020 annotations. For each instance, we showed annotators the image with the highlighted region and the top-5 candidate descriptions from the respective method (Figure E.1). Annotators indicated which candidates (if any) were related to the region—either *concretely* (directly visible), *abstractly* (conceptually related), or *globally* (present elsewhere in image). Each instance was annotated by at least one author, with an average of 1.8 annotators per instance.

A visual token is labeled *interpretable* if a majority of annotators selected at least one candidate as related. Comparing human majority vote against the LLM judge, we find substantial agreement: Cohen's $\kappa = 0.68$ and accuracy = 84.4% across all 1,020 instances.

Table E.1: Ablation results showing LATENTLENS interpretability change and nearest-neighbor overlap with baseline, averaged across all layers. Δ **Interp.**: Change in % interpretable (baseline = 72.3%). **Top-5 NN Overlap**: Average number of matching neighbors out of 5. Token = same subword; Phrase = same full caption. Captioning ablations maintain interpretability with partial overlap (∼2/5), suggesting multiple valid mappings exist.

		Top-5 NN Overlap	
Model Variant	**Δ Interp.**	**Token**	**Phrase**
Baseline (OLMo + ViT-L)	72.3%	—	—
Different seed	+0.4%	2.5	2.2
Linear connector	−0.2%	2.1	2.0
First-sentence captions	−2.5%	1.8	1.5
Unfrozen LLM	+5.5%	1.9	1.5
Spatial Task (frozen)	−34.2%	0.0	0.0
Spatial Task (unfrozen)	−30.2%	0.0	0.0

E.4 Ablations

In this subsection we ablate several components of our training setup to determine to what extent our findings depend on our particular setup. In other words, will a given image patch be mapped to the same nearest-neighbour words, regardless of training data, task, or mapping function? Specifically, if we do find similar LATENTLENS interpretations of the same input across ablations, it could imply that visual tokens represent something akin to raw task-independent input and all the task-specific extraction and computation is conducted by the LLM.

Experimental setup. We focus on OLMo + CLIP-ViT and ablate the following 5 aspects:

1. The **random seed** used for initializing the connector weights and controlling the training data sampling.

2. Using a **linear mapping** instead of 3-layer MLP to map visual tokens into

the LLM prefix space.

3. Varying the **level of detail** in the captioning dataset by training on single-sentence captions instead of the detailed multi-sentence Pixmo-Cap dataset.

4. **Unfreezing the LLM** during training. When allowing the LLM weights to adapt to the task of processing and describing visual tokens, two outcomes are plausible: the proportion of interpretable tokens either drops or it rises. For the former, we can conceive that some of the LLM weights specialize to be a visual processing model (and not a language model), with less training pressure to represent tokens as words a frozen LLM can process. On the other hand, we can also imagine that now the connector module simply became more expressive, being able to align any visual token to LLM embeddings, even if the initial structure of both embedding space was sometimes non-trivially different.

5. **Changing the task** from captioning to a spatial prediction task. The model is trained to answer the question "Where is [OBJECT]?" with either "top" or "bottom". The hypothesis we test is whether the language-based task of captioning leads to the observed alignment of visual tokens to the LLM embedding space.

For each ablation we measure two metrics, averaged across all layers: (1) **LatentLens Interpretability**: The percentage of visual tokens judged interpretable by GPT-5, reported as change from baseline (72.3%). (2) **LatentLens Overlap**: Average number of matching top-5 LatentLens nearest neighbors with the baseline model (out of 5). We report both *token-level* overlap (same subword in both top-5 sets) and *phrase-level* overlap (same full contextual phrase).

Interpretability is robust to training variations but requires language-based objectives. Results are summarized in Table E.1. Simply changing the seed results in only around half of the top-5 LatentLens results to overlap with

the original run. All of them are still highly similar concepts but it is some indication that even the seed can have some influence, albeit it is unclear how semantically meaningful it is. On the other hand, we find encouraging results when replacing the 3-layer MLP connector with a single matrix (linear layer) and when replacing highly detailed captions as training data with single-sentence captions: The level of interpretability is almost the same, and we still see a significant amount of overlap between the original top-5 nearest neighbors (2.1 and 1.8 out of 5, respectively). Thus, the relationship between vision encoder and LLM embeddings spaces can be linearly aligned in an interpretable manner and moreover, short captioning data is enough to do so[1]. Next, we observe a notable increase in the amount of interpretable visual tokens (+5.5%) when unfreezing the LLM during training. Thus, with this more "expressive connector" in the form of some LLM weights, the model can learn a more interpretable alignment. Finally, we observe that the language-based task of captioning is necessary for interpretable visual tokens. When instead training the model to generate a single token ("top" or "bottom") given a question about the location of an object, interpretability drops by around 30%, with zero overlap to the original LATENTLENS top-5 NNs. The tokens that are still marked as interpretable for this spatial task are the same few generic words such as "upper", "background", or "outside".

Training Dynamics We investigate *when* LATENTLENS interpretability emerges during connector training. We re-ran the OLMo-7B + CLIP-ViT connector from scratch, saving intermediate checkpoints at steps 100, 1000, and 6000 (in addition to the final step 12000), and evaluated LATENTLENS on each using our LLM judge.

Results are shown in Table E.2. At step 100 (0.8K training examples), interpretability is near-random (~12%) across all layers. By step 1000 (8K examples, ~8% of training), we find an interesting difference across layers: the earliest LLM

[1]Follow-up work could explore what the limit of simplicity of the training data is for this interpretable alignment to appear.

layers (0–1) already reach 67% interpretability, while deeper layers (2–31) remain at only 17%. By step 6000 (48K examples, half of training), all layers converge to near-final performance. This suggests that alignment first forms at the input of the LLM — where visual tokens enter the residual stream — and only later propagates to deeper representations.

Table E.2: Training dynamics of LATENTLENS interpretability (OLMo+CLIP-ViT). Average % of interpretable visual tokens across 9 LLM layers, plus the split between early layers (0–1) and deeper layers (2–31). Interpretability emerges first in the earliest layers at step 1000, while all layers reach near-final performance by step 6000.

Step	Examples	Avg	Layers 0–1	Layers 2–31
100	0.8K	11.7%	13.3%	11.3%
1000	8K	28.1%	**67.0%**	17.0%
6000	48K	75.1%	75.4%	75.0%
12000	96K	71.3%	71.0%	71.4%

E.5 Tuned Lens Comparison

Belrose et al. (2023) propose *Tuned Lens*, which learns a lightweight per-layer residual affine probe to improve upon LogitLens. Each probe transforms the intermediate representation $\mathbf{h}$ as $\hat{\mathbf{h}} = \mathbf{h} + \mathbf{W}\mathbf{h} + \mathbf{b}$ (where $\mathbf{W}$ is initialized to zero so the probe starts as an identity map), and the result is then decoded with the unembedding matrix (the same final step as LogitLens). The probes are trained to minimize the KL divergence between the probe's output distribution and the final-layer output distribution: $\mathscr{L} = \mathrm{KL}(p_{\text{final}} \,\|\, q_{\text{probe}})$.

Implementation. We follow the Belrose et al. (2023) recipe as close as possible. We use SGD with Nesterov momentum (lr=0.1, momentum=0.9, weight_decay=10^{-3}), a linear learning-rate schedule, and gradient clipping (clip=1.0). The weight decay regularizes each probe toward the identity transformation. We train the probes on visual token positions only, using 2,000 PixMoCap images (≈1.15M visual tokens total), one probe per layer. We select

four model pairs: the three lowest-LogitLens-scoring pairs from our main evaluation (LLaMA3+SigLIP 7.1%, LLaMA3+DINOv2 7.2%, Qwen2+SigLIP 10.8%) and the highest (OLMo+CLIP 34.3%), spanning the full range of LogitLens performance.

Table E.3: Tuned Lens vs. LogitLens and LATENTLENS (average % interpretable across layers). Tuned Lens provides little or no improvement over LogitLens on visual tokens, while LATENTLENS outperforms both by a large margin on every model.

Model pair	LogitLens	Tuned Lens	LATENTLENS
LLaMA3-8B + SigLIP	7.1	8.9	62.3
LLaMA3-8B + DINOv2	7.2	6.4	77.4
Qwen2-7B + SigLIP	10.8	7.4	74.3
OLMo-7B + CLIP	34.3	25.2	72.3

Results. Tuned Lens yields at best a marginal gain (+1.8 pp on LLaMA3+SigLIP) and is worse than LogitLens on the other three pairs. On OLMo+CLIP, the model where LogitLens already works reasonably well, early-layer probes do improve interpretability, but this benefit reverses at later layers (a collapse of 31–44 pp at layers 24–31), despite the identity-initialised probes and ℓ_2 weight decay that were designed to prevent exactly this kind of drift. The result is a net decrease of -9.1 pp relative to plain LogitLens.

E.6 Vision vs. Text Token L2 Norms

Figure E.2 shows the L2 norm distributions for all 9 model combinations (3 LLMs $\times$ 3 vision encoders). For each model, we show separate histograms for visual tokens (left) and text tokens (right), colored by layer (yellow=early, red=late). Key observations:

- **Vision tokens have larger L2 norms than text tokens** across all models, often by 1–2 orders of magnitude.

- **OLMo-7B** maintains relatively small L2 norms (max ≈1000) for both modalities.
- **LLaMA3-8B and Qwen2-7B** exhibit much larger L2 norms for visual tokens, with max values exceeding 100,000 in some cases.
- **DINOv2** consistently produces the largest L2 norms across all LLMs.
- The 99th percentile (p99, black dotted line) and maximum (red dashed line) markers show substantial outliers in visual token distributions.

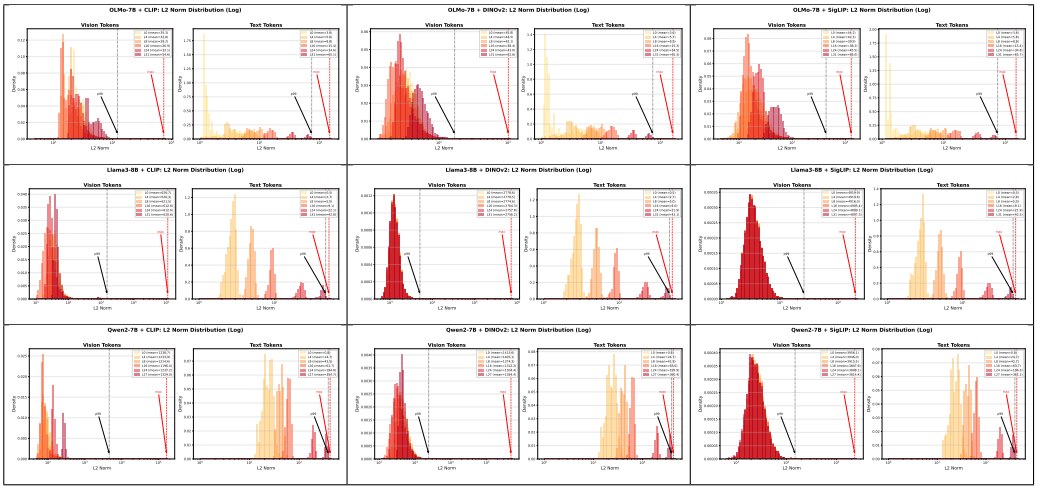


Figure E.2: L2 norm distributions of vision vs. text tokens across layers. Each row corresponds to an LLM (OLMo-7B, LLaMA3-8B, Qwen2-7B), and each column shows a vision encoder (CLIP, DINOv2, SigLIP). Within each cell, Vision tokens (left) and Text tokens (right) are shown. Colors indicate layer depth (yellow=early, red=late). The x-axis uses log scale. Black dotted lines mark the 99th percentile; red dashed lines mark the maximum value.

Are High L2 Norms from Sparse Outliers or Uniform Scaling? To understand whether high L2 norms are driven by a few large embedding dimensions (sparse outliers) or uniformly larger values across all dimensions, we extract the full embedding vector of the visual token with maximum L2 norm for each model and analyze its distribution.

Figure E.3 shows histograms of individual embedding dimension values for these max-L2-norm tokens. The key finding is striking:

- **All distributions are approximately Gaussian**, not sparse. High L2 norms come from uniformly larger values across *all* 3584–4096 dimensions, not from a few extreme outliers.
- **OLMo-7B has ~100× smaller embedding scale** than LLaMA3-8B and Qwen2-7B. Standard deviations are ~10–20 for OLMo vs. ~1700–11000 for LLaMA3/Qwen2.
- **LLaMA3-8B max tokens occur at layer 0** (input), while OLMo and Qwen2 max tokens occur at **layer 24** (late layers).

This suggests that the MLP connector (or already the vision encoder itself) learns fundamentally different embedding scales depending on the target LLM architecture, with LLaMA3 and Qwen2 resulting in much larger magnitude projections than OLMo.

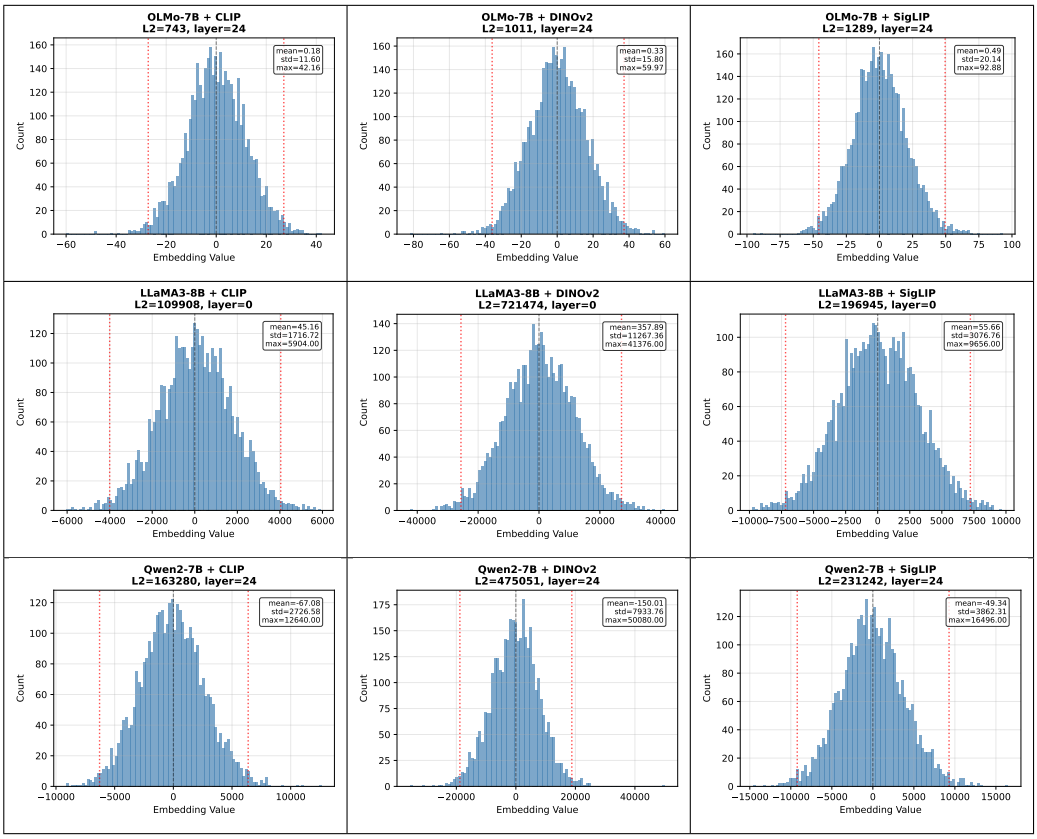


Figure E.3: Distribution of embedding dimension values for max L2 norm visual tokens. Each row corresponds to an LLM (OLMo-7B, LLaMA3-8B, Qwen2-7B), and each column shows a vision encoder. We extract the visual token with the largest L2 norm and plot a histogram of its individual embedding dimension values. All distributions are Gaussian-like, indicating that high L2 norms result from uniform scaling across all dimensions rather than sparse outliers.

E.7 Cosine Similarity Distributions

Figure E.4 shows the distribution of top-1 nearest neighbor cosine similarities between visual tokens and their closest contextual text embeddings at layer 16 (a representative mid-layer). Several model combinations exhibit **bimodal distributions** with two distinct peaks. This pattern is most pronounced for LLaMA3-8B + CLIP and Qwen2-7B + CLIP, where one peak appears around

0.15–0.2 and a second around 0.35–0.4. A similar bimodal pattern is visible for Qwen2-7B + DINOv2.

In contrast models paired with SigLIP tend to show more unimodal distributions. The bimodal structure suggests that visual tokens may fall into two categories: those that align well with text representations (higher similarity) and those that remain more distant from the text embedding space (lower similarity). This could correspond to tokens representing salient visual content versus background or less semantically meaningful regions.

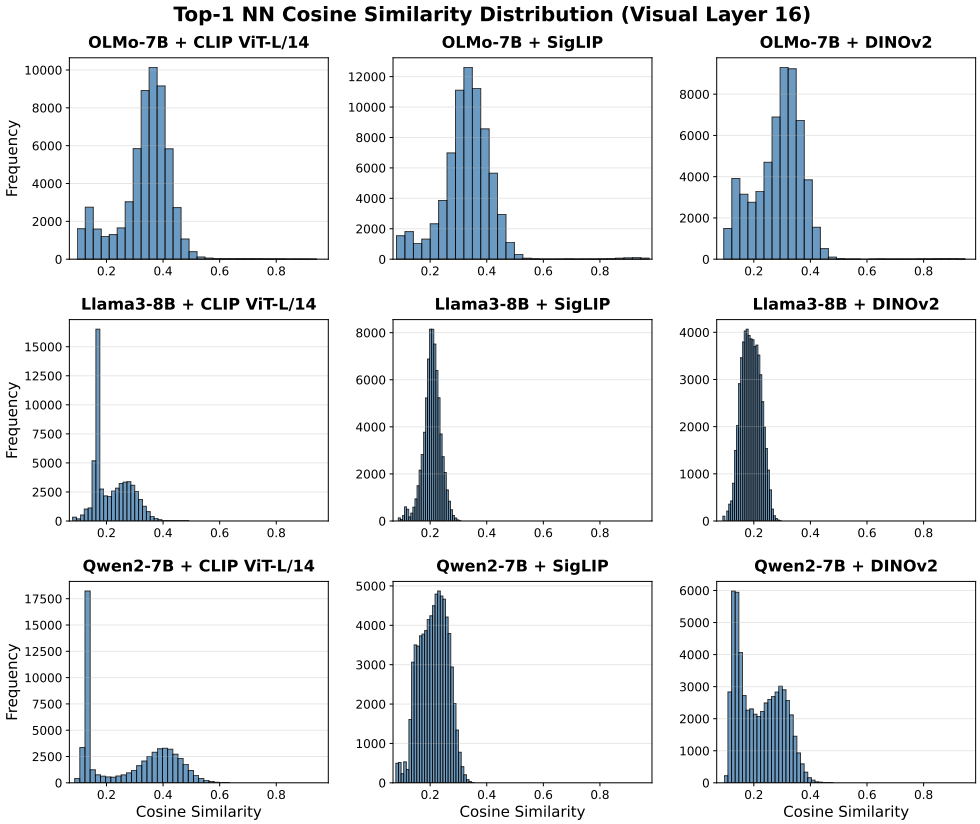


Figure E.4: Distribution of top-1 nearest neighbor cosine similarities at layer 16. Each panel shows the histogram of cosine similarities between visual tokens and their closest contextual text embedding for one LLM–vision encoder combination. Several combinations (notably LLaMA3 + CLIP, Qwen2 + CLIP) exhibit bimodal distributions, suggesting distinct populations of visual tokens with different degrees of alignment to the text embedding space.

E.8 Layer Alignment Details

To understand why visual tokens at the input layer align most strongly with contextual text representations from *later* layers (the Mid-Layer Leap, Section 5.4.3), we investigate how much visual token representations change as they pass through the LLM.

Figure E.5 shows the cosine similarity between each token at layer ℓ and

its own representation at layer 0, averaged across all tokens. For text tokens, similarity to the input embedding drops rapidly within the first few layers (often below 0.4 by layer 4), reflecting the contextualization process where tokens incorporate information from surrounding context. In contrast, visual tokens maintain much higher similarity to their input representations—often above 0.8 through mid-layers—indicating they undergo substantially less transformation during LLM processing.

This minimal drift of visual tokens explains the Mid-Layer Leap: since visual tokens are already "pre-contextualized" by the vision encoder and connector, they arrive at the LLM in a representational state more similar to how the LLM represents text after several layers of processing. The frozen LLM then processes these tokens with relatively little modification, as evidenced by the high layer-to-layer similarity.

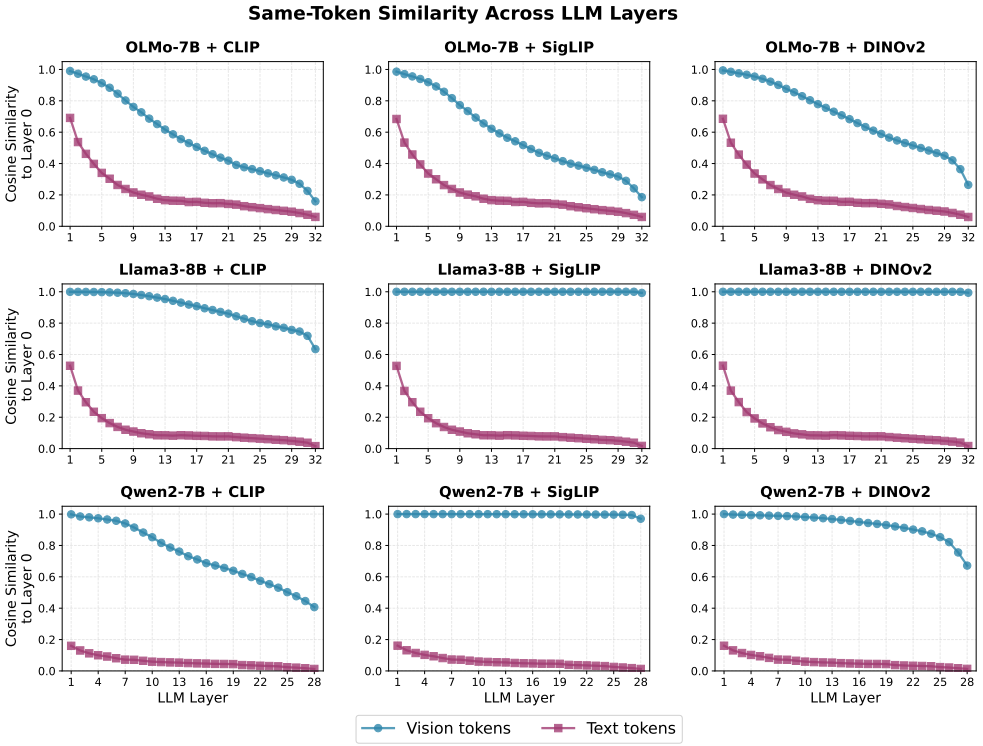
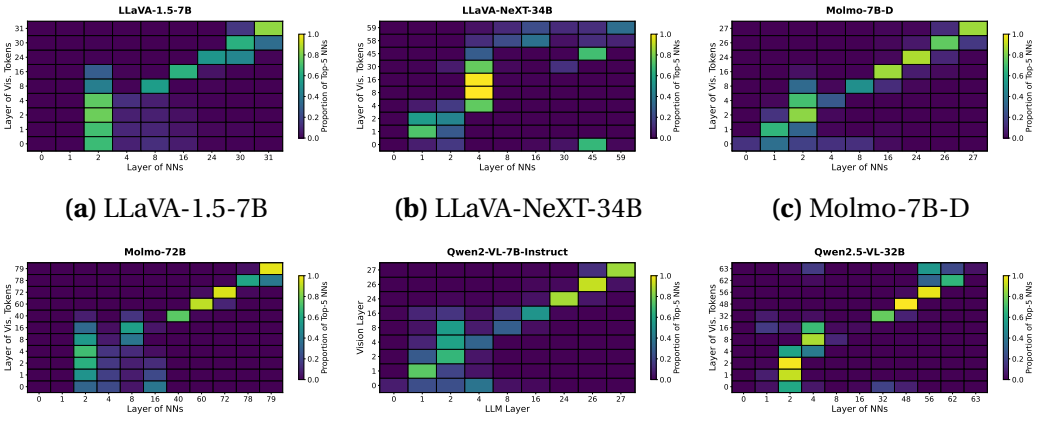


Figure E.5: Vision tokens follow only minor drift through LLM processing: We compare the same token (e.g. same position in the image or text passage) to its input-layer embedding across layers. We find that text tokens early on have little similarity with their initial embeddings, perhaps following some process of contextualization, abstraction, or simply preparing for next-token prediction. On the other hand, visual tokens display a much higher cosine similarity to their input embeddings, especially until middle layers.

E.9 Results for off-the-shelf VLMs

This section provides additional details for Section 5.4.4. We verify that the *Mid-Layer Leap* generalizes to all six off-the-shelf VLMs, and provide a deeper analysis for Qwen2-VL-7B-Instruct.

Mid-Layer Leap across all six models Figure E.6 shows the layer alignment heatmaps for all six off-the-shelf VLMs. All six exhibit the same hallmark pattern



(a) LLaVA-1.5-7B **(b)** LLaVA-NeXT-34B **(c)** Molmo-7B-D

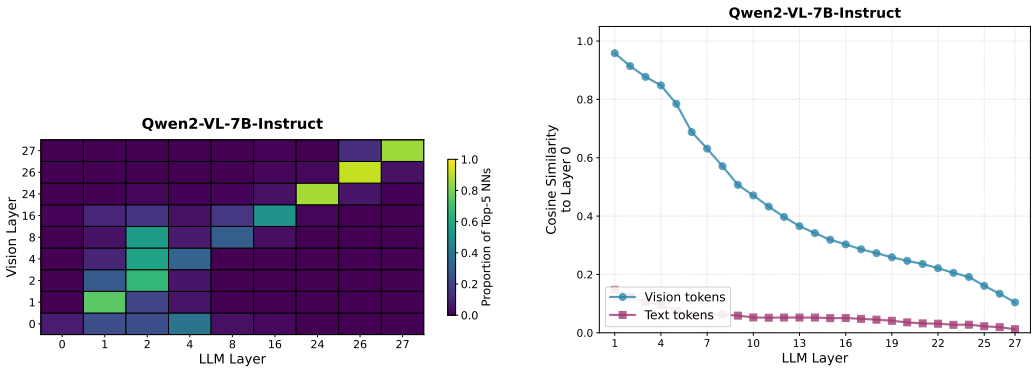
(d) Molmo-72B **(e)** Qwen2-VL-7B **(f)** Qwen2.5-VL-32B

Figure E.6: Mid-Layer Leap in six off-the-shelf VLMs. Layer alignment heatmaps showing which LLM layer's contextual embeddings are most similar to visual tokens at each processing stage. All six models show the same pattern as the controlled setup: a leap at layer 0 toward middle layers, then a diagonal alignment from mid-processing onward.

observed in the controlled setup (Figure 5.4): visual tokens at layer 0 align primarily to middle LLM layers (the leap), while from mid-processing onward a diagonal pattern emerges. This confirms that the Mid-Layer Leap is not an artifact of our controlled training setup but a robust property of how visual tokens interact with LLM representations.

Deeper analysis: Qwen2-VL-7B-Instruct Among the six off-the-shelf models, we conduct a more detailed analysis for Qwen2-VL-7B-Instruct (Wang et al., 2024a) as a representative example.

Setup Qwen2-VL differs from our controlled training along several axes: it was trained in multiple stages on 1.4 trillion tokens; crucially the LLM and vision encoder are unfrozen at some stages; it uses various different datasets at different stages, e.g. instruction tuning data; and finally it relies on elaborate image token preprocessing. We apply the same evaluation methodology as before: 100 random visual token patches evaluated by our LLM judge across



(a) Layer alignment **(b)** Token drift

Figure E.7: Qwen2-VL layer analysis. (a) Which LLM layer's contextual embeddings are most similar to visual tokens at each processing stage. (b) Cosine similarity of tokens at each layer to their input-layer representation. Visual tokens undergo substantial transformation (0.96→0.10), while text tokens start low (0.15) and stay low—unlike frozen LLMs where visual tokens retain higher similarity.

layers $\ell \in \{0, 1, 2, 4, 8, 16, 24, 26, 27\}$. Since Qwen2-VL uses a finetuned version of Qwen2's LLM (not the frozen base model), we extract contextual embeddings from Qwen2-VL's own LLM backbone to ensure proper alignment.

Mid-Layer Leap Figure E.7 shows the layer alignment and token drift for Qwen2-VL. For visual tokens at layer 0, the leap is to LLM layer 4, then from layer 1 onward a clear diagonal pattern emerges. Visual tokens also undergo more change throughout LLM layers than in our frozen-LLM controlled setup, though still substantially less than text tokens (see Figure E.5).

E.10 Fine-grained Interpretation Analysis

Beyond measuring *whether* visual tokens are interpretable (Section 5.4), we analyze *what kinds* of interpretations LATENTLENS produces. We examine three dimensions: (1) interpretation types—whether nearest neighbors describe something concrete, abstract, or global (based on LLM judge outputs); (2) parts-of-speech—the grammatical categories of matched words (using spacy from

Honnibal et al. (2020)); and (3) visual attributes—how often color, shape, and texture words appear. Key findings: concrete interpretations dominate (65–75%), nouns are most common (45–50%), and color words decline from early to late layers while other ratios remain stable—suggesting the frozen LLM does not progressively abstract away from concrete visual content.

E.10.1 Interpretation Types

Figure E.8: Breakdown of interpretation types. Top row: categories (Concrete 65%, Abstract 19%, Global 16%). Second row: top most frequent nearest-neighbor words per category. Lower rows: example Visual Genome phrases showing context, with target word in **bold**. Data aggregated across all 10 models (9 controlled setups plus Qwen2-VL).

Figure E.8 provides a visual breakdown of interpretation types aggregated across all models. Figure E.9 shows the breakdown for all 9 model combinations individually. We categorize LATENTLENS interpretations as:

- **Concrete**: The nearest neighbor word directly names something visible in the image region (objects, colors, textures).
- **Abstract**: The word describes a concept related to but not literally visible (emotions, activities, functions).
- **Global**: The word describes something present elsewhere in the image but not in the highlighted region.

Across all models and layers, concrete interpretations dominate (70–75% on average), with abstract and global each contributing 11–15%. The ratios remain remarkably stable across layers, suggesting that the frozen LLM does not progressively "abstract away" from concrete visual descriptions.

As a representative off-the-shelf model, Figure E.10 shows the same analysis for Qwen2-VL-7B-Instruct (see Section E.9 for further details on this model). The pattern is similar: concrete interpretations dominate (62–78%), though we observe a slight decrease in later layers (26–27: 62%) with a corresponding increase in abstract interpretations. This may reflect Qwen2-VL's unfrozen LLM having learned some layer-wise abstraction.

Figure E.9: Interpretation types for all 9 model combinations. Each bar shows the breakdown of interpretable tokens into concrete (directly visible), abstract (conceptually related), and global (present elsewhere in image). Concrete interpretations dominate across all models and layers (70–90%). SigLIP models show notably higher global interpretations (20–30%), consistent with less localized nearest neighbors.

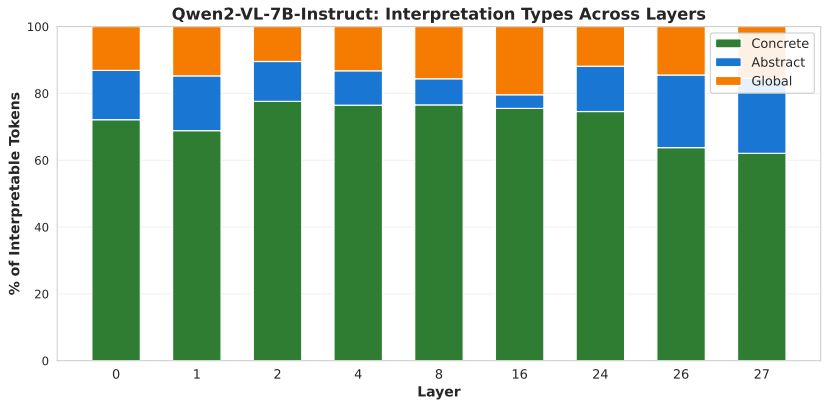


Figure E.10: Interpretation types for Qwen2-VL-7B-Instruct. Similar to the frozen LLM models, concrete interpretations dominate. A slight decrease in concrete (and increase in abstract) is observed in later layers (26–27), possibly reflecting the unfrozen LLM's learned abstraction.

E.10.2 Parts-of-Speech and Visual Attributes

Figure E.11 shows the distribution of parts-of-speech among all nearest neighbors for each of the 9 trained model combinations. Nouns dominate at approximately 45–50%, which aligns with the visual nature of the interpretations—visual tokens primarily encode objects and entities. Proper nouns account for 10–20%, verbs 10–15%, and adjectives around 5%. The distribution is relatively stable across layers, with some variation between models (e.g., OLMo-7B + ViT-L shows more variability). Figure E.12 shows the same analysis for Qwen2-VL-7B-Instruct, which exhibits similar patterns.

Figure E.13 shows the frequency of visual attribute words (colors, shapes, textures) for each trained model. Color words are the most common at around 5–6% in early layers, declining to around 3% in later layers. This suggests that raw color information is more prominent in early visual token representations. Shape and texture words are rare throughout (<1%). Figure E.14 shows the same for Qwen2-VL.

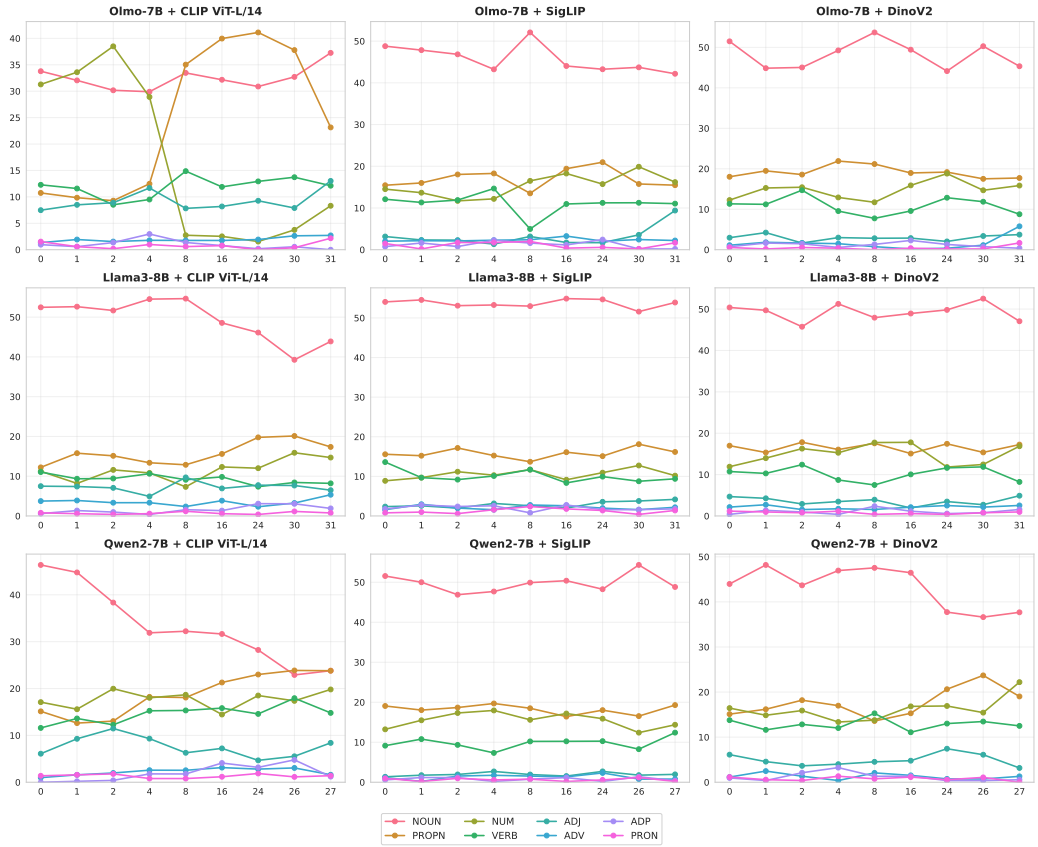


Figure E.11: Parts-of-speech distribution across layers for 9 trained models. Nouns dominate (45–50%), followed by proper nouns and verbs. The distribution is relatively stable across layers with some per-model variation.

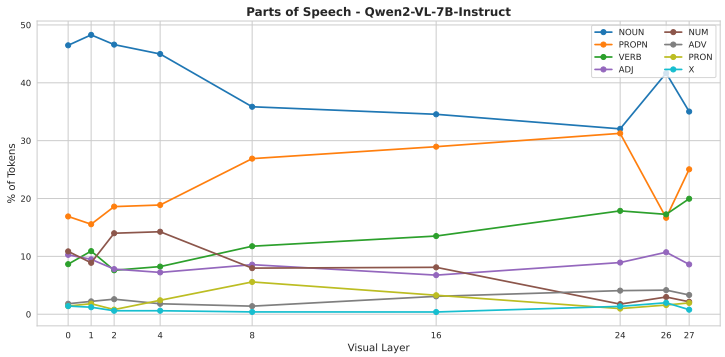


Figure E.12: Parts-of-speech distribution for Qwen2-VL-7B-Instruct. Similar pattern to trained models: nouns dominate, followed by proper nouns and verbs.

Figure E.13: Visual attribute word frequency for 9 trained models. Color words are common (5–6% early, declining to 3% late), while shape and texture words are rare (<1%).

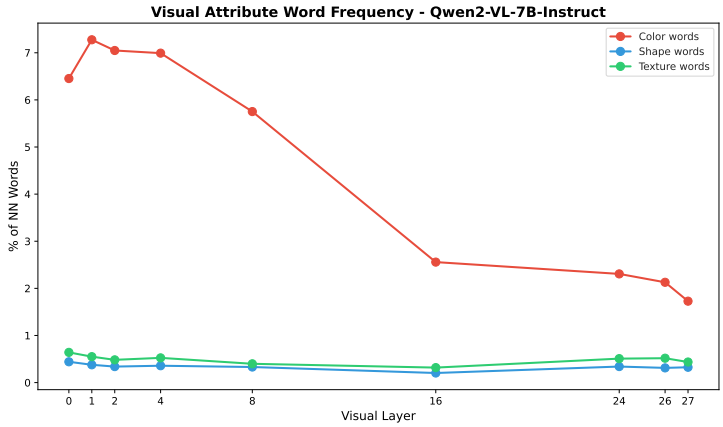


Figure E.14: Visual attribute word frequency for Qwen2-VL-7B-Instruct. Similar pattern: color words dominate visual attributes, shape and texture are rare.

E.11 Phrase-Level Interpretation Examples

Quantifying the value of context. To measure how much the preceding context contributes to interpretation quality, we randomly sample 50 examples where the LLM judge deemed the top LATENTLENS result interpretable. For each example, we manually label whether the preceding context helps interpretation compared to the word alone, using three categories: *yes* (context clearly aids understanding), *neutral* (context neither helps nor hurts), or *no* (context is misleading or irrelevant).

For instance, comparing "large stone tower with gold **clocks**" to just "**clocks**"—the former provides richer spatial and material context that better matches the visual content.

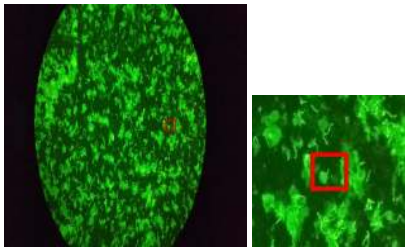
We find that in **64%** of cases, the preceding context provides a better interpretation than the word alone. In 28% of cases the context was neutral (the word alone was sufficient), and in only 8% was the context misleading. This validates that LATENTLENS's phrase-level interpretations offer meaningful advantages over token-level approaches.

Qualitative examples. We show 12 randomly selected (non-cherry-picked) examples from our phrase annotation study. Each panel shows the visual token's patch (red box) with preprocessing matching the respective vision encoder (CLIP preserves aspect ratio with padding, SigLIP and DINOv2 squash to square). Below each image we show: the LATENTLENS phrase versus a random Visual Genome phrase containing the same token. The highlighted word shows the matched token, demonstrating how contextual information can sometimes aid interpretation.

LatentLens: “pals : the pair were spotted posing in matching person - branded aprons at the event , with a large blue car parked **behind** them”

Random phrase, same token: “the trees are **behind** the giraffe”

LLaMA3+DINOv2, L4



LatentLens: “black eye with white **specks** around it”

Random phrase, same token: “ **rocks** beside zebra”

LLaMA3+SigLIP, L30

LatentLens: “woman wearing black sweatpants and grey **long-sleeve** t-shirt”

Random phrase, same token: “photo was taken by jenny **lee** silver”

LLaMA3+CLIP, L16

LatentLens: “the blue shirt the **bald** man has on.”

Random phrase, same token: “smiling **bald** man”

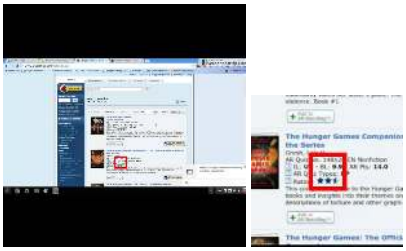
OLMo+DINOv2, L4

Figure E.15: Phrase annotation examples (1–4). Each panel shows a vision token’s patch (red box) preprocessed as the vision encoder sees it. **LatentLens**: Top contextual nearest neighbor phrase from LATENTLENS. **Random**: A random VG phrase containing the same token.

LatentLens: “multiple types of cherry tomatoes in multiple **colours** , all in bluegreen boxes”

Random phrase, same token: “something black w/ wild **colours** deflated before the tent; hot air balloon, maybe?”

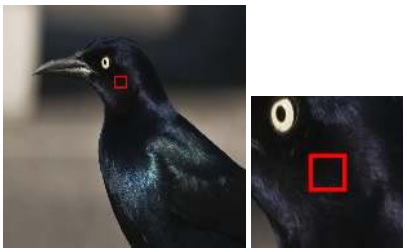
OLMo+SigLIP, L4



LatentLens: “five **stars** on the side of the bus”

Random phrase, same token: “the **stars** below the plane.”

OLMo+CLIP, L30



LatentLens: “the cow is mostly **black** ”

Random phrase, same token: “the non **black** car on the roadway.”

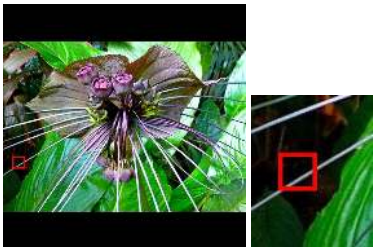
Qwen2+DINOv2, L8

LatentLens: “man wearing white short sleeve **tunic** ”

Random phrase, same token: “two **unicorns** dancing together”

Qwen2+SigLIP, L0

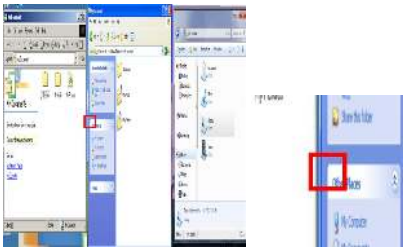
Figure E.16: Phrase annotation examples (5–8).



LatentLens: “green leaves are in the **background** .”

Random phrase, same token: “the green vegetation in the **background** ”

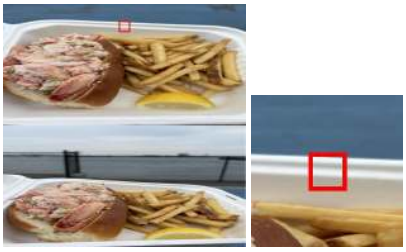
Qwen2+CLIP, L24



LatentLens: “bowl is white and has two **sections** ”

Random phrase, same token: “dried splinted wood **sections** ”

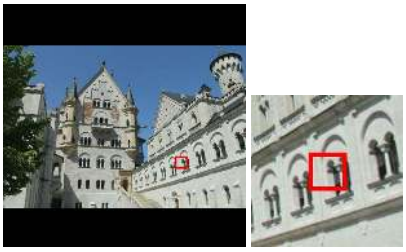
LLaMA3+DINOv2, L16



LatentLens: “barbecue sauce on the side of a **styrofoam** cup”

Random phrase, same token: “a metal **rooster** on a pole”

LLaMA3+SigLIP, L1



LatentLens: “a pale coloured wooden model of an old house with pitched roof and gable **windows** stands on a table .”

Random phrase, same token: “row of three **windows** ”

LLaMA3+CLIP, L16

Figure E.17: Phrase annotation examples (9–12).

E.12 Dynamic Corpus Generation

As noted in Section 5.5, our fixed corpus contains at most 20 contextual embeddings per vocabulary token. We investigate whether *dynamically generating* phrase contexts—rather than relying on a fixed corpus—can yield better LA-

TENTLENS interpretations.

Method. We use an evolutionary search approach: given a visual token and its top-5 LATENTLENS nearest neighbors from the fixed corpus, we iteratively generate variations using GPT-4o and keep those with highest cosine similarity to the visual embedding. Crucially, since LLMs are autoregressive, we only modify words *before* the target token (the token whose contextual embedding we extract), keeping the target token at the end of each phrase.

Specifically, we run 6 rounds of evolution with 20 variations per round, keeping the top-5 phrases. We evaluate on 20 visual tokens that our LLM judge marked as interpretable (OLMo-7B + CLIP ViT-L/14, layer 16).

Results. Dynamic generation improves cosine similarity in 85% of cases (17/20), with an average improvement of +0.017. Table E.4 shows representative examples. The evolved phrases tend to be more concise and visually specific—for instance, "a white and black peaked **building** with a peaked roof" (similarity 0.415) evolves to "grand arched beige **building**" (similarity 0.463).

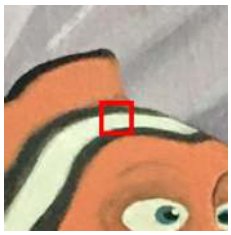
Original (VG corpus)	**Evolved (dynamic)**	**Δsim**
shiny red apples with bright green **leaves**	colorful stamen against green **leaves**	+0.052
a white and black peaked **building** with a peaked roof	grand arched beige **building**	+0.047
the sign reads dried **beef** king	the placard shows assorted **meats**	+0.039
a forest of **lush** trees	bright canopy of **treetops**	+0.031
a nice clear blue **sky**	glistening blue **sky**	+0.023

Table E.4: Examples of dynamic phrase generation improving LATENTLENS interpretations. The target token is shown in **bold**. Evolved phrases achieve higher cosine similarity to the visual embedding by finding more appropriate contextual framing. Note that evolution can also discover better target tokens (e.g., "beef" → "meats").

Interestingly, in 35% of cases (7/20), tokens that were not the top-1 match in the original corpus rose to become the best match after evolution. For exam-

ple, when the original top-5 contained multiple candidate tokens (“building”, “facade”, “mansion”), the evolutionary search sometimes found that a different token with better surrounding context outperformed the original top-1. This suggests that the fixed corpus’s limitation of 20 sentences per token may cause some tokens to be underrepresented due to suboptimal context, even when they would be semantically fitting.

Here, we illustrate three levels of descriptions and their richness obtained from: 1) dynamically generated context, 2) sentences from a fixed corpus, and 3) same token but with lowest scoring context:



Dynamic: “colorful fish with white **stripes** ”
(cosine similarity: 0.36)

Fixed Corpus: “man wearing white **striped** [...]”
(cosine similarity: 0.34)

Lowest: “white stripe above two blue **stripes** ”
(cosine similarity: 0.26)

E.13 Captioning Quality Evaluation

We evaluate caption quality using DCScore (Ye et al., 2025), a GPT-4o-based LLM judge that rates generated captions on a 1–10 scale across 300 validation images from PixMo-Cap.

Evaluation rubric. DCScore evaluates each caption on four fine-grained criteria, each scored 1–10:

- **Faithfulness**: How accurately the caption reflects the actual image content
- **Detail accuracy**: Coverage and correctness of salient visual details
- **Hallucinations**: Absence of non-existent content (higher = fewer hallucinations)
- **Completeness**: Whether the caption captures all key aspects of the image

The overall score is a holistic quality rating that considers all criteria. We provide the judge with both the full image and the generated caption, asking it to return a structured JSON response with all sub-scores.

Full results. Table E.5 shows the complete caption quality scores for all 9 trained model combinations plus the off-the-shelf Qwen2-VL-7B-Instruct as an upper bound reference.

Table E.5: Caption quality scores (DCScore, 1–10 scale) on 300 PixMo-Cap validation images. Higher is better. Our trained models average 6.0, while the fully instruction-tuned Qwen2-VL achieves 8.5.

LLM	Vision Encoder	Score
OLMo-7B	CLIP ViT-L/14	6.60
OLMo-7B	SigLIP	6.75
OLMo-7B	DINOv2-Large	4.30
LLaMA3-8B	CLIP ViT-L/14	6.79
LLaMA3-8B	SigLIP	6.95
LLaMA3-8B	DINOv2-Large	4.46
Qwen2-7B	CLIP ViT-L/14	7.08
Qwen2-7B	SigLIP	6.77
Qwen2-7B	DINOv2-Large	4.54
Upper bound: Qwen2-VL-7B-Instruct		**8.50**

Key observations. **(1)** DINOv2 models consistently underperform on captioning (4.3–4.5) compared to CLIP and SigLIP models (6.6–7.1), likely due to DINOv2's lack of language supervision during pretraining. **(2)** Qwen2-7B achieves the highest average score (6.13) across vision encoders among our trained models. **(3)** The gap between our models (avg 6.0) and Qwen2-VL (8.5) reflects that our models were trained only on captioning with a frozen LLM, while Qwen2-VL underwent full multimodal instruction tuning. **(4)** Notably, captioning performance does not correlate with visual token interpretability—DINOv2 models show comparable interpretability (Figure 5.3) despite lower captioning scores.

Sample captions. Figure E.18 shows example captions from three representative models on the same validation image, illustrating how different model combinations produce captions of varying quality and detail.

OLMo + CLIP-ViT (8/10)

The image depicts a bustling square, likely in Italy, characterized by a mix of old and new architecture. Dominating the scene is a grand, ancient building with a distinctive yellow facade, adorned with a clock tower and a prominent cross. The building features a series of arches and a grand entrance, flanked by two large columns. The square is bustling with people, some seated on benches, others strolling around.

Qwen2 + SigLIP (8/10)

The image depicts a vibrant and bustling square in Italy, characterized by a mix of historical and modern architecture. The central focus is a large, ornate fountain, which is the focal point of the square. To the left of the fountain, there is a building with a distinctive clock tower adorned with statues. The sky above is a clear blue, enhancing the overall beauty of the scene.

LLaMA3 + DINOv2 (7/10)

The image captures a bustling square in a European city, likely in Italy or Spain, with a vibrant mix of architectural styles. Dominating the scene is a large, ornate building with a golden facade and a prominent clock tower. Adjacent to the clock tower is a smaller building with a similar golden hue. In the foreground, a large fountain draws a crowd of people.

Figure E.18: Sample captions for the same validation image. Per-image DCScores shown in parentheses.

E.14 Qualitative examples

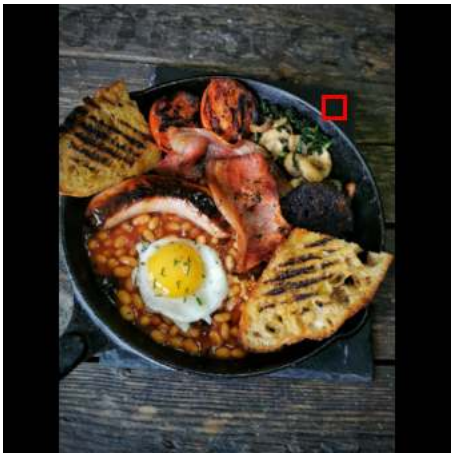
To demonstrate that our findings are not cherry-picked, we present 20 randomly sampled examples across 5 layers (0, 8, 16, 24, and final layer). For each example, we randomly select one of 10 models (9 trained models plus Qwen2-VL) and show top-3 predictions from each method: **EmbeddingLens** (nearest neighbors in input embedding matrix), **LogitLens** (LM head predictions), and **LATENTLENS** (ours, contextual nearest neighbors with phrase context).

For LATENTLENS, we show the top-3 Visual Genome phrases containing the matched token (highlighted in **yellow**), demonstrating the semantic rich-

ness of phrase-level interpretations. For baselines, we show the top-3 tokens with EmbeddingLens and LogitLens colored backgrounds. Across all randomly sampled examples, LATENTLENS consistently provides more interpretable results—the contextual phrases typically describe the visual content more accurately than isolated tokens from the baselines.

OLMo+SigLIP, L0

LatentLens:
(1) “bear’s paws covered in **swirling** grass”
(2) “blue tail of plane with white **swirls** ”
(3) “the cat’s tail is **frizzy** .”
EmbeddingLens: hairy | Skinny | flashy
LogitLens: cir | faucet | rawn

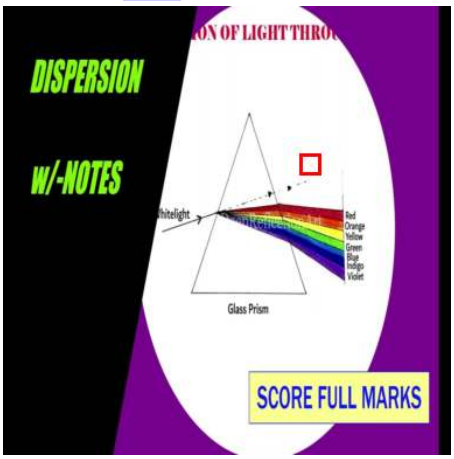


LLaMA3+CLIP, L0

LatentLens:
(1) “a vase sitting on a square **table** ”
(2) “water glass on a white **tablecloth** ”
(3) “cup and saucer on a wooden **table** ”
EmbeddingLens: planted | opies | avou
LogitLens: [?] | uzey | tối

OLMo+SigLIP, L0

LatentLens:
(1) “train with the letters s. **w.n** . in white”
(2) “a sign reading "carrer d’en **falconer** ”"
(3) “train with the letters **s.w** .n. in white”
EmbeddingLens: Vanessa | [?] | Wolf
LogitLens: hausen | ens | experiment..



OLMo+SigLIP, L0

LatentLens:
(1) “the water had the sun **gleaming** down”
(2) “...ncluding a **dissassembled** bicycle and...”
(3) “bathing suit for swimming and **suntanning** .”
EmbeddingLens: [HI] | 218 | Education
LogitLens: erne | rone | crest

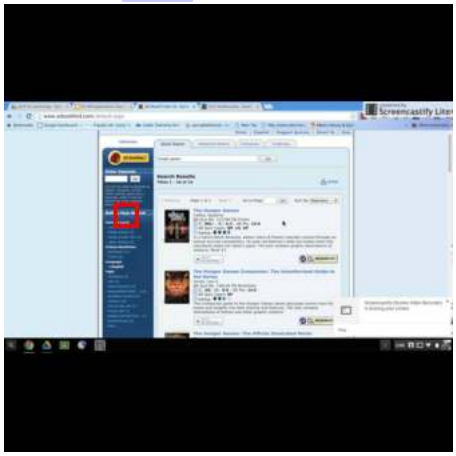
Figure E.19: Layer 0 (input): Four randomly sampled visual tokens at layer 0.

Qwen2+DINOv2, L8

LatentLens:
(1) "...s, probably 3 [**though** the one beneat..."
(2) "a second adult **bows** low in to inspect t..."
(3) "...e bear+person [**though** that might not..."
EmbeddingLens: -c [?] (translate..
LogitLens: tô Según ==========..

Qwen2-VL, L8

LatentLens:
(1) "controls are on the center **console** "
(2) "dials on motorcycle **dashboard** glow red"
(3) "controls are on the center **console** "
EmbeddingLens: [TH] .urlencoded .DropDownL..
LogitLens: [?] <nav Jihad



Qwen2+CLIP, L8

LatentLens:
(1) "a **wikipedia** screen is displayed on the ..."
(2) "a few **paragraphs** written on the screen"
(3) "... real [check **wikipedia**]"
EmbeddingLens: "‘ .DrawString incorrect
LogitLens: stantiate chia IsValid

LLaMA3+CLIP, L8

LatentLens:
(1) "sky with two large **willowy** clouds"
(2) "baby blue sky with **wispy** cirrus clouds"
(3) "...red with thin **branches** with green le..."
EmbeddingLens: Schultz 95 Vert
LogitLens: strstr Kron [?]

Figure E.20: Layer 8 (early-mid): Four randomly sampled visual tokens at layer 8.

Qwen2+CLIP, L16

LatentLens:
(1) “stone chimney attached to a **house** ”
(2) “pole and trees in front of **building** ”
(3) “low, grey roof of **residence** .”
EmbeddingLens: (translate.. ([... .isRequired
LogitLens: chantment [?] onUpdate

Qwen2+SigLIP, L16

LatentLens:
(1) “framed pencil **sketch** on wall”
(2) “pencil **sketched** artwork of a fruit basket”
(3) “framed **drawing** to left of the bed”
EmbeddingLens: There div -cut
LogitLens: [?] [?] fdc

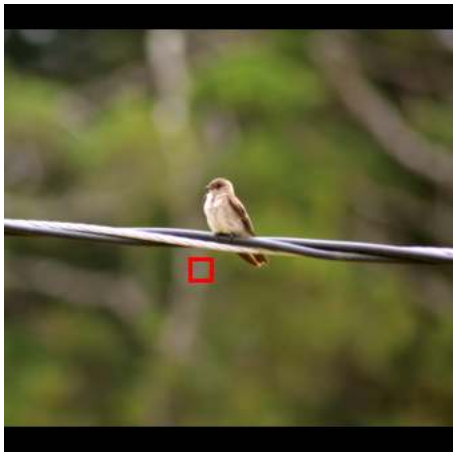
Qwen2+DINOv2, L16

LatentLens:
(1) “...ith black polka **dots** ”
(2) “...er with a blue **stipe** ”
(3) “man wearing a white shirt with blue **stipe** ”
EmbeddingLens: .COMP Specifies rà
LogitLens: CascadeType =nil oppon

OLMo+CLIP, L16

LatentLens:
(1) “man with balding head **standing** near cart”
(2) “...diners, four of **whom** are toasting, a...”
(3) “older man **sitting** on wooden bench”
EmbeddingLens: him men Boys
LogitLens: minority mostly oya

Figure E.21: Layer 16 (middle): Four randomly sampled visual tokens at layer 16.



LLaMA3+CLIP, L24

LatentLens:
(1) "...leaves against **muted** background"
(2) "...hink black and **white** stripes"
(3) "...st a wall with **muted** white lights ."
EmbeddingLens: white | White | allied
LogitLens: .scalablyt.. | UnitOfWork | [?]

Qwen2+SigLIP, L24

LatentLens:
(1) "man wearing **white** baseball cap."
(2) "lady in **off-white** sweater"
(3) "back of woman wearing a white **turtleneck** "
EmbeddingLens: .leading | machining | GetX
LogitLens: _unregister | [?] | ("/

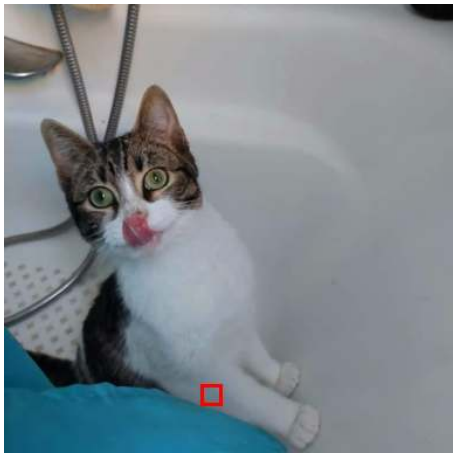
LLaMA3+DINOv2, L24

LatentLens:
(1) "a woman in a tan **blazer** "
(2) "man wearing grey **coat** "
(3) "woman wears an **off-white** ski jacket"
EmbeddingLens: o | k | Drawable
LogitLens: classpath | [?] | [?]

LLaMA3+CLIP, L24

LatentLens:
(1) "... lime green , **apricot** orange , turqu..."
(2) "blue, **bergundy** and gray coat."
(3) "tan/ **yellow** dog laying across young girls lap"
EmbeddingLens: Gold | gold | white
LogitLens: GOLD | ConverterF.. | HEST

Figure E.22: Layer 24 (late): Four randomly sampled visual tokens at layer 24.

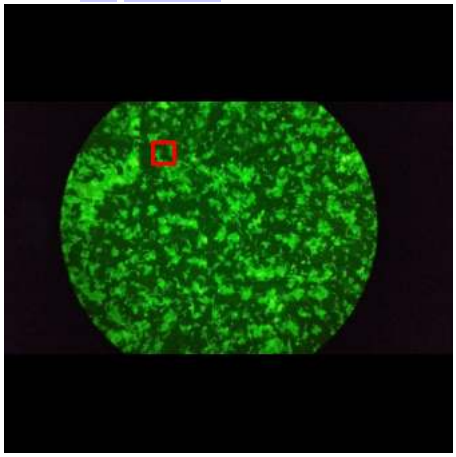


LLaMA3+SigLIP, L31

LatentLens:
(1) "...in a degree of **filth,** or at least gr..."
(2) "beneath it ' **fahrradstrae',** capital 'f',..."
(3) "...ot a passenger **plane,** but a plane fo..."
EmbeddingLens: [?] Yard x
LogitLens: [?] _defaults mia

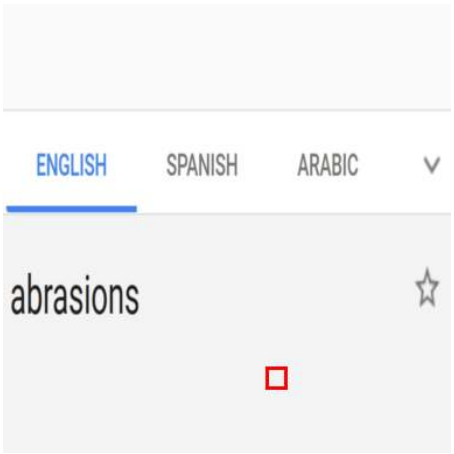
OLMo+CLIP, L31

LatentLens:
(1) " **wall-mounted** , white box and toilet pape..."
(2) " **wall-mounted** mirror, with decorative, b..."
(3) " **wall-mounted** flat screen television"
EmbeddingLens: quil Cl %;
LogitLens: speakers speaker .



Qwen2+CLIP, L27

LatentLens:
(1) "the black **tilted** wheel"
(2) "a **tilted** black pole"
(3) "a **tilted** cookie sheet"
EmbeddingLens: fileName [?] elapsed
LogitLens: -left left right



Qwen2+SigLIP, L27

LatentLens:
(1) "window on the side of the **conference** room"
(2) "the **app** for watching youtube videos"
(3) "the **app** for watching vimeo videos"
EmbeddingLens: [TH] [?] ,
LogitLens: [?] <?= [?]

Figure E.23: Final layer: Four randomly sampled visual tokens at the final layer (31 for OLMo/LLaMA, 27 for Qwen2).

E.15 Behind The Scenes

The goal of this section is to make science more transparent and engaging, showing not just the polished paper at the end but also all the detours and lessons learned.

E.15.1 From start to finish

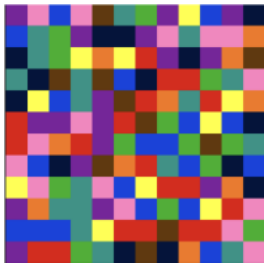
The pivot. This project started last winter (December 2024) when the first author, originally working on vision-and-language models in general, was determined to pivot towards *understanding* models and fundamental science. So interpretability was the natural direction to look into, from afar it had been intriguing to follow the field the past few years. But just doing interpretability for the sake of interpretability did not feel like the right approach. So what was an actual fundamental question that many people would care about, ourselves included, where interpretability could naturally help?

Brainstorming. The first author tried to present too many potential ideas at once in their lab meeting. There was not enough time to present them all, and it would not have been a fun presentation to follow with too many disjoint pitches. So a lab colleague simply asked: "Which one of these questions are you genuinely excited about?" And this is how we ended up with this paper and the research question we ask: How can frozen LLMs possibly make sense of visual tokens? Do they look like language tokens to the model? We started with this simple question but the first author could not have possibly imagined all these new insights we would stumble upon, such is the process of science. It was really a situation of unknown unknowns: the kind of experiments and ideas we would eventually end up with were inconceivable at the time, and how with each new experiment, three new options opened up. As a result writing the paper in 8 pages was very challenging and three sections had to be cut even from the appendix.

Don't trust assumptions. If the first author had to take away just one lesson from the project, it would be to never trust long-held assumptions (either by oneself or the field). We had assumed that visual tokens were not really interpretable with nearest neighbors from the embedding matrix, and several papers hinted in that direction. In retrospect, we should have simply tested this empirically across several models from day one. Instead, the first author read papers about anisotropy and other interpretability literature, prematurely concluding that embeddings from different modalities live in entirely different narrow cones.[2] So, then the question became: If these tokens are not interpretable and live in different subspaces, maybe we have to perform more linear algebra and embedding space tricks to show how they relate to text? Rotate them around, learn some simple mappings, specific subspaces, ...?

The Mosaic Dataset. Eventually, the first author hypothesized: Maybe visual tokens are not directly mapped to interpretable words (measured by NNs from the embedding matrix) because real data is messy? Visual objects in a scene span many tokens, or a single token might represent several objects and different attributes. So it would be no surprise that they don't just cleanly map to a single concept. But what if we could simplify this situation? What if we could make the data we train on simpler and simpler until we see very predictable nearest neighbors. This is where the project went "off-track" for a while and we started experimenting with toy datasets we coined *Mosaic*, where the model gets as input an image like this:

[2] We even ran lots of experiments on anisotropy and concluded that effects reported in other papers seemed overly simplified. At this point I wouldn't feel confident saying that different modalities live in different narrow cones inside e.g. an LLM.



The LLM would then be trained to predict the sequence of color words one after the other ("red blue red pink ..."), essentially "copying" each visual token's corresponding color to the text space. We observed some interesting quirks, but surprisingly not many interpretable nearest neighbors (i.e. color words from the embedding matrix). In fact, fewer than on the natural images we report in this paper. We even ran various causal patching experiments, studied high-norm outliers, etc! They all seemed interesting at the time, but again, simply testing our assumptions early would have saved us these detours. At this point we had already adopted the Molmo codebase for training models, since we cared about how *frozen* LLMs make sense of visual tokens. But most VLMs you can take off-the-shelf unfreeze the LLM weights at some stage of training, which is why we opted for our controlled training setup that the final paper still builds upon.

Hope. The first author does not remember when exactly we finally questioned our assumptions, and empirically tested the interpretability on natural images. But sometime around July, we started exploring interactive demos of natural images with the top NN from the embedding matrix. Since we happened to explore OLMo+CLIP-ViT, we were surprised to see so many meaningful words. The cosine similarity was low (between e.g. 0.07 and 0.15), much lower than what LATENTLENS eventually yields. Nonetheless this was encouraging, but it soon became clear that automating the judgment of whether an NN is interpretable is not trivial.[3]

[3] It would take another 3 months to develop the full LLM judge we eventually relied upon.

A first glimpse of the final story. With these exciting findings (40% to 60% of tokens in OLMo+CLIP-ViT are interpretable), the initial story of the paper was roughly: Visual tokens at layer 0 are more interpretable than some would assume! In the background we kept working on directions that did not end up in the main paper or even appendix: 1) We conducted ablations under which conditions this interpretability would increase or decrease, now in Section E.4, though without any trace of these initial EmbeddingLens results. 2) We always wondered what these strange non-interpretable tokens encode... They often had the same EmbeddingLens nearest neighbors, they were often in non-salient background regions. Were they something akin to task vectors or register tokens? At the time this seemed like an interesting enough story to publish: Past work assumes visual tokens are rarely interpretable at the input via EmbeddingLens but here we show for some models that is not the case. We show ablations, we show patching experiments. Some qualitative results. It is good we kept exploring further: Some co-authors kept wondering, especially SR and MM: What happens to visual tokens at LLM layers after the input? And that's where the final paper we now have started taking shape. We adopted EmbeddingLens and LogitLens not just at the beginning or end of the model but throughout. And eventually we wondered: What if we use contextual embeddings instead of static embedding matrices? (Thank you MM and Elinor Poole-Dayan for the nudge!)

E.15.2 Lessons and Reflections

Automation. This project co-occurred with the rise of Cursor and Claude Code. Especially interactive demos for quick exploration are now much easier to build, crucial for projects like this one. Due to the speed of experimentation, a lot of ideas on the side did not make it into the paper but can now be easily continued as follow-up work.

Lessons on science. As mentioned above, testing the simplest assumptions early on is something the first author will keep as a guiding principle for future

research. The field knows less than one would expect. Models change constantly, experiments might look different with slightly different setups. Re-run what others have seemingly done before.

Interpretability is fun. The process of open-ended discovery is very enjoyable and naturally leads to interesting brainstorming sessions with colleagues and friends. The first author highly recommends working at least on one interpretability project in one's research journey. The community is quite unique, reflective and open to good ideas (ideally) regardless of whether they are immediately useful downstream. The first author recently wrote a blog post, which goes into detail about how the pivot to interpretability last year went: Better late than never: Getting into interpretability in 2025.

Personal. This project carries a lot of meaning. It will be the final chapter of the first author's PhD and was in many ways a perfect ending to this five-year long journey. I have rarely felt so proud of a work, and again not just because of the final product, but the whole process: Every single collaborator on here was fantastic, either as my closest friends throughout the PhD or as recent cherished connections and mentors. A special shout out goes to MM and DE who both put a lot of care into this paper. This project also perfectly encapsulates the first author's strengthened conviction to stay in academia for the long haul. In a way it represented what academia ideally could be and what we should strive for it to be. Looking back at other projects, never before did I have so many enjoyable interactions when sharing what we are working on. Even before the work is now getting out, I gave four talks about it (three in-person), and many more Mila coffee table chats. It made me realize how science is so much more than papers, and papers are just one way to let others know what you found. One can give talks, write blog posts, make videos, tell others about your work, write it into the snow, showcase it as a demo.

References

Vaibhav Adlakha, Parishad BehnamGhader, Fabian David Schmidt, Nicolas Chapados, Marius Mosbach, and Siva Reddy. 2026. LLM2Vec-Gen: Generative embeddings from large language models. *arXiv preprint arXiv:2603.10913*.

Aishwarya Agrawal, Dhruv Batra, and Devi Parikh. 2016. Analyzing the behavior of visual question answering models. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1955–1960.

Aishwarya Agrawal, Jiasen Lu, Stanislaw Antol, Margaret Mitchell, C. Lawrence Zitnick, Devi Parikh, and Dhruv Batra. 2015. Vqa: Visual question answering. *International Journal of Computer Vision*, 123:4–31.

Saba Ahmadi, Rabiul Awal, Ankur Sikarwar, Amirhossein Kazemnejad, Ge Ya Luo, Juan Rodriguez, Sai Rajeswar Mudumba, Siva Reddy, Chris Pal, Benno Krojer, et al. 2026. The promise of rl for autoregressive image editing. *Advances in Neural Information Processing Systems*, 38:117510–117545.

Betty van Aken, Benjamin Winter, Alexander Löser, and Felix A Gers. 2020. Visbert: Hidden-state visualizations for transformers. In *Companion Proceedings of the Web Conference 2020*, pages 207–211.

Mubashara Akhtar, Anka Reuel, Prajna Soni, Sanchit Ahuja, Pawan Sasanka Ammanamanchi, Ruchit Rawal, Vilém Zouhar, Srishti Yadav, Chenxi Whitehouse, Dayeon Ki, Jennifer Mickel, Leshem Choshen, Marek Šuppa, Jan Batzner, Jenny Chim, Jeba Sania, Yanan Long, Hossein A. Rahmani, Christina Knight, Yiyang Nan, Jyoutir Raj, Yu Fan, Shubham Singh, Subramanyam Sahoo, Eliya Habba, Usman Gohar, Siddhesh Pawar, Robert Scholz, Arjun Subramonian, Jingwei Ni, Mykel Kochenderfer, Sanmi Koyejo, Mrinmaya Sachan, Stella Biderman, Zeerak Talat, Avijit Ghosh, and Irene Solaiman. 2026. When ai benchmarks plateau: A systematic study of benchmark saturation.

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. 2022. Flamingo: a visual language model for few-shot learning. *Advances in neural information processing systems*, 35:23716–23736.

Hadi Alzayer, Zhihao Xia, Xuaner Zhang, Eli Shechtman, Jia-Bin Huang, and Michael Gharbi. 2024. Magic fixup: Streamlining photo editing by watching dynamic videos. *arXiv preprint arXiv:2403.13044*.

Jacob Andreas and Dan Klein. 2016. Reasoning about Pragmatics with Neural Listeners and Speakers. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1173–1182, Austin, Texas. Association for Computational Linguistics.

Jacob Andreas, Marcus Rohrbach, Trevor Darrell, and Dan Klein. 2016. Neural module networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 39–48.

Mohammad Asadi, Jack W O'Sullivan, Fang Cao, Tahoura Nedaee, Kamyar Rajabalifardi, Fei-Fei Li, Ehsan Adeli, and Euan Ashley. 2026. Mirage: The illusion of visual understanding. *arXiv preprint arXiv:2603.21687*.

Mido Assran, Adrien Bardes, David Fan, Quentin Garrido, Russell Howes, Matthew Muckley, Ammar Rizvi, Claire Roberts, Koustuv Sinha, Artem Zholus, et al. 2025. V-jepa 2: Self-supervised video models enable understanding, prediction and planning. *arXiv preprint arXiv:2506.09985*.

Rabiul Awal, Saba Ahmadi, Le Zhang, and Aishwarya Agrawal. 2024. Vismin: Visual minimal-change understanding. *Advances in Neural Information Processing Systems*, 37:107795–107829.

Jinze Bai, Shuai Bai, Shusheng Yang, Shijie Wang, Sinan Tan, Peng Wang, Junyang Lin, Chang Zhou, and Jingren Zhou. 2023. Qwen-vl: A frontier large vision-language model with versatile abilities. *arXiv preprint arXiv:2308.12966*.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, et al. 2025. Qwen2.5-VL Technical Report. *arXiv preprint arXiv:2502.13923*.

Luke Bailey, Gustaf Ahdritz, Anat Kleiman, Siddharth Swaroop, Finale Doshi-Velez, and Weiwei Pan. 2023. Soft prompting might be a bug, not a feature. In *ICML 2023 Workshop on DeployableGenerativeAI*.

Ankan Bansal, Yuting Zhang, and Rama Chellappa. 2020. Visual question answering on image sets. In *ECCV*.

Hritik Bansal, Zongyu Lin, Tianyi Xie, Zeshun Zong, Michal Yarom, Yonatan Bitton, Chenfanfu Jiang, Yizhou Sun, Kai-Wei Chang, and Aditya Grover. 2025. Videophy: Evaluating physical commonsense for video generation. In *The Thirteenth International Conference on Learning Representations*.

Yonatan Belinkov. 2022. Probing classifiers: Promises, shortcomings, and advances. *Computational Linguistics*, 48(1):207–219.

Nora Belrose, Zach Furman, Logan Smith, Danny Halawi, Igor Ostrovsky, Lev McKinney, Stella Biderman, and Jacob Steinhardt. 2023. Eliciting latent predictions from transformers with the tuned lens. *arXiv preprint arXiv:2303.08112*.

Lucas Beyer, Andreas Steiner, André Susano Pinto, Alexander Kolesnikov, Xiao Wang, Daniel Salz, Maxim Neumann, Ibrahim Alabdulmohsin, Michael Tschannen, Emanuele Bugliarello, et al. 2024. Paligemma: A versatile 3b vlm for transfer. *arXiv preprint arXiv:2407.07726*.

Mahtab Bigverdi, Zelun Luo, Cheng-Yu Hsieh, Ethan Shen, Dongping Chen, Linda G Shapiro, and Ranjay Krishna. 2025. Perception tokens enhance visual reasoning in multimodal language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 3836–3845.

Abeba Birhane, Vinay Uday Prabhu, and Emmanuel Kahembwe. 2021. Multimodal datasets: misogyny, pornography, and malignant stereotypes. *arXiv preprint arXiv:2110.01963*.

Yonatan Bisk, Ari Holtzman, Jesse Thomason, Jacob Andreas, Yoshua Bengio, Joyce Chai, Mirella Lapata, Angeliki Lazaridou, Jonathan May, Aleksandr Nisnevich, Nicolas Pinto, and Joseph Turian. 2020a. Experience grounds language. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8718–8735, Online. Association for Computational Linguistics.

Yonatan Bisk, Rowan Zellers, Ronan Le Bras, Jianfeng Gao, and Yejin Choi. 2020b. PIQA: Reasoning about physical commonsense in natural language. In *Proceedings of the Thirty-Fourth AAAI Conference on Artificial Intelligence*, volume 34, pages 7432–7439.

Kevin Black, Mitsuhiko Nakamoto, Pranav Atreya, Homer Rich Walke, Chelsea Finn, Aviral Kumar, and Sergey Levine. 2024. Zero-shot robotic manipulation with pre-trained image-editing diffusion models. In *The Twelfth International Conference on Learning Representations*.

Ben Bogin, Shivanshu Gupta, Matt Gardner, and Jonathan Berant. 2021. COVR: A test-bed for visually grounded compositional generalization with real images. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 9824–9846, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Florian Bordes, Quentin Garrido, Justine T Kao, Adina Williams, Michael Rabbat, and Emmanuel Dupoux. 2025. Intphys 2: Benchmarking intuitive physics understanding in complex synthetic environments. *arXiv preprint arXiv:2506.09849*.

Andrew Brock, Jeff Donahue, and Karen Simonyan. 2019. Large scale GAN training for high fidelity natural image synthesis. In *International Conference on Learning Representations*.

Tim Brooks, Aleksander Holynski, and Alexei A Efros. 2023. Instructpix2pix: Learning to follow image editing instructions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18392–18402.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Emanuele Bugliarello, Ryan Cotterell, Naoaki Okazaki, and Desmond Elliott. 2021. Multimodal Pretraining Unmasked: A Meta-Analysis and a Unified Framework of Vision-and-Language BERTs. *Transactions of the Association for Computational Linguistics*, 9:978–994.

Ryan Burgert, Kanchana Ranasinghe, Xiang Li, and Michael S Ryoo. 2022. Peekaboo: Text to image diffusion models are zero-shot segmentors. *arXiv preprint arXiv:2211.13224*.

Aylin Caliskan, Joanna J Bryson, and Arvind Narayanan. 2017. Semantics derived automatically from language corpora contain human-like biases. *Science*, 356(6334):183–186.

Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. 2020. End-to-end object detection with transformers. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part I 16*, pages 213–229. Springer.

David M Chan, Rodolfo Corona, Joonyong Park, Cheol Jun Cho, Yutong Bai, and Trevor Darrell. 2024. Analyzing the language of visual tokens. *arXiv preprint arXiv:2411.05001*.

Huiwen Chang, Han Zhang, Jarred Barber, AJ Maschinot, Jose Lezama, Lu Jiang, Ming-Hsuan Yang, Kevin Murphy, William T Freeman, Michael Rubinstein, et al. 2023. Muse: Text-to-image generation via masked generative transformers. *arXiv preprint arXiv:2301.00704*.

Ting-Yun Chang and Yun-Nung Chen. 2019. What does this word mean? explaining contextualized embeddings with natural language definition. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 6064–6070.

Haozhe Chen, Junfeng Yang, Carl Vondrick, and Chengzhi Mao. 2024. INViTE: INterpret and control vision-language models with text explanations. In *The Twelfth International Conference on Learning Representations*.

Tianle Chen, Chaitanya Chakka, Arjun Reddy Akula, Xavier Thomas, and Deepti Ghadiyaram. 2026. Some modalities are more equal than others: Decoding and architecting multimodal integration in mllms. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2142–2151.

Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. 2020a. Uniter: Universal image-text representation learning. In *European Conference on Computer Vision*, pages 104–120. Springer.

Yen-Chun Chen, Linjie Li, Licheng Yu, Ahmed El Kholy, Faisal Ahmed, Zhe Gan, Yu Cheng, and Jingjing Liu. 2020b. UNITER: UNiversal Image-TExt Representation Learning. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision – ECCV 2020*, volume

12375, pages 104–120. Springer International Publishing, Cham. Series Title: Lecture Notes in Computer Science.

Jaemin Cho, Abhay Zala, and Mohit Bansal. 2022. Dall-eval: Probing the reasoning skills and social biases of text-to-image generative transformers. *arXiv preprint arXiv:2202.04053*.

Jang Hyun Cho, Andrea Madotto, Effrosyni Mavroudi, Triantafyllos Afouras, Tushar Nagarajan, Muhammad Maaz, Yale Song, Tengyu Ma, Shuming Hu, Suyog Jain, et al. 2025. Perceptionlm: Open-access data and models for detailed visual understanding. *arXiv preprint arXiv:2504.13180*.

Yunfei Chu, Jin Xu, Qian Yang, Haojie Wei, Xipin Wei, Zhifang Guo, Yichong Leng, Yuanjun Lv, Jinzheng He, Junyang Lin, et al. 2024. Qwen2-audio technical report. *arXiv preprint arXiv:2407.10759*.

Christopher Clark, Jieyu Zhang, Zixian Ma, Jae Sung Park, Mohammadreza Salehi, Rohun Tripathi, Sangho Lee, Zhongzheng Ren, Chris Dongjoo Kim, Yinuo Yang, et al. 2026. Molmo2: Open weights and data for vision-language models with video understanding and grounding. *arXiv preprint arXiv:2601.10611*.

Herbert H. Clark. 1996. *Using Language*. Cambridge University Press.

Kevin Clark and Priyank Jaini. 2023. Text-to-image diffusion models are zero-shot classifiers. In *ICLR 2023 Workshop on Mathematical and Empirical Understanding of Foundation Models*.

Reuben Cohn-Gordon, Noah Goodman, and Christopher Potts. 2018. Pragmatically informative image captioning with character-level inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 439–443, New Orleans, Louisiana. Association for Computational Linguistics.

Arthur Conmy, Augustine Mavor-Parker, Aengus Lynch, Stefan Heimersheim, and Adrià Garriga-Alonso. 2023. Towards automated circuit discovery for mechanistic interpretability. *Advances in Neural Information Processing Systems*, 36:16318–16352.

Guillaume Couairon, Jakob Verbeek, Holger Schwenk, and Matthieu Cord. 2023. Diffedit: Diffusion-based semantic image editing with mask guidance. In *The Eleventh International Conference on Learning Representations*.

Hoagy Cunningham, Aidan Ewart, Logan Riggs, Robert Huben, and Lee Sharkey. 2023. Sparse autoencoders find highly interpretable features in language models. *arXiv preprint arXiv:2309.08600*.

Dima Damen, Hazel Doughty, Giovanni Maria Farinella, Sanja Fidler, Antonino Furnari, Evangelos Kazakos, Davide Moltisanti, Jonathan Munro, Toby Perrett, Will Price, et al. 2018. Scaling egocentric vision: The epic-kitchens dataset. In *Proceedings of the European conference on computer vision (ECCV)*, pages 720–736.

Abhishek Das, Satwik Kottur, Khushi Gupta, Avi Singh, Deshraj Yadav, José MF Moura, Devi Parikh, and Dhruv Batra. 2017. Visual dialog. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 326–335.

Pradipto Das, Chenliang Xu, Richard F Doell, and Jason J Corso. 2013. A thousand frames in just a few words: Lingual description of videos through latent topics and sparse object stitching. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2634–2641.

Matt Deitke, Christopher Clark, Sangho Lee, Rohun Tripathi, Yue Yang, Jae Sung Park, Mohammadreza Salehi, Niklas Muennighoff, Kyle Lo, Luca Soldaini, et al. 2024. Molmo and pixmo: Open weights and open data for state-of-the-art multimodal models. *arXiv preprint arXiv:2409.17146*.

Matt Deitke, Christopher Clark, Sangho Lee, Rohun Tripathi, Yue Yang, Jae Sung Park, Mohammadreza Salehi, Niklas Muennighoff, Kyle Lo, Luca Soldaini, et al. 2025. Molmo and pixmo: Open weights and open data for state-of-the-art vision-language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 91–104.

Chaorui Deng, Deyao Zhu, Kunchang Li, Chenhui Gou, Feng Li, Zeyu Wang, Shu Zhong, Weihao Yu, Xiaonan Nie, Ziang Song, Guang Shi, and Haoqi Fan. 2025. Emerging properties in unified multimodal pretraining.

Roberto Dessì, Eleonora Gualdoni, Francesca Franzon, Gemma Boleda, and Marco Baroni. 2022. Communication breakdown: On the low mutual

intelligibility between human and neural captioning. *arXiv preprint arXiv:2210.11512*.

Benjamin Devillers, Léopold Maytié, and Rufin VanRullen. 2024. Semi-supervised multimodal representation learning through a global workspace. *IEEE Transactions on Neural Networks and Learning Systems*, 36(5):7843–7857.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Nelson Elhage, Neel Nanda, Catherine Olsson, Tom Henighan, Nicholas Joseph, Ben Mann, Amanda Askell, Yuntao Bai, Anna Chen, Tom Conerly, Nova DasSarma, Dawn Drain, Deep Ganguli, Zac Hatfield-Dodds, Danny Hernandez, Andy Jones, Jackson Kernion, Liane Lovitt, Kamal Ndousse, Dario Amodei, Tom Brown, Jack Clark, Jared Kaplan, Sam McCandlish, and Chris Olah. 2021. A mathematical framework for transformer circuits. *Transformer Circuits Thread*.

Kawin Ethayarajh. 2019. How contextual are contextualized word representations? Comparing the geometry of BERT, ELMo, and GPT-2 embeddings. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 55–65, Hong Kong, China. Association for Computational Linguistics.

Matan Eyal, Shoval Sadde, Hillel Taub-Tabib, and Yoav Goldberg. 2022. Large scale substitution-based word sense induction. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4738–4752.

Constanza Fierro, Negar Foroutan, Desmond Elliott, and Anders Søgaard. 2025. How do multilingual language models remember facts? In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 16052–16106, Vienna, Austria. Association for Computational Linguistics.

Jerry Fodor. 2001. Language, thought and compositionality. *Royal Institute of Philosophy Supplements*, 48:227–242.

Maxwell Forbes, Christine Kaeser-Chen, Piyush Sharma, and Serge Belongie. 2019. Neural Naturalist: Generating Fine-Grained Image Comparisons. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 708–717, Hong Kong, China. Association for Computational Linguistics.

Stephanie Fu, tyler bonnen, Devin Guillory, and Trevor Darrell. 2025. Hidden in plain sight: VLMs overlook their visual representations. In *Second Conference on Language Modeling*.

Tsu-Jui Fu, Wenze Hu, Xianzhi Du, William Yang Wang, Yinfei Yang, and Zhe Gan. 2023. Guiding instruction-based image editing via multimodal large language models. *ArXiv*, abs/2309.17102.

Xingyu Fu, Yushi Hu, Bangzheng Li, Yu Feng, Haoyu Wang, Xudong Lin, Dan Roth, Noah A Smith, Wei-Chiu Ma, and Ranjay Krishna. 2024. Blink: Multimodal large language models can see but not perceive. In *European Conference on Computer Vision*, pages 148–166. Springer.

Yossi Gandelsman, Alexei A Efros, and Jacob Steinhardt. 2024. Interpreting CLIP's image representation via text-based decomposition. In *The Twelfth International Conference on Learning Representations*.

Quentin Garrido, Nicolas Ballas, Mahmoud Assran, Adrien Bardes, Laurent Najman, Michael Rabbat, Emmanuel Dupoux, and Yann LeCun. 2025. Intuitive physics understanding emerges from self-supervised pretraining on natural videos.

Manu Gaur, Darshan Singh S, and Makarand Tapaswi. 2025. No detail left behind: Revisiting self-retrieval for fine-grained image captioning. *Transactions on Machine Learning Research*.

Timnit Gebru, Jamie Morgenstern, Briana Vecchione, Jennifer Wortman Vaughan, Hanna Wallach, Hal Daumé Iii, and Kate Crawford. 2021. Datasheets for datasets. *Communications of the ACM*, 64(12):86–92.

Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A Wichmann. 2020. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673.

Mor Geva, Jasmijn Bastings, Katja Filippova, and Amir Globerson. 2023. Dissecting recall of factual associations in auto-regressive language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12216–12235, Singapore. Association for Computational Linguistics.

Mor Geva, Avi Caciularu, Kevin Wang, and Yoav Goldberg. 2022. Transformer feed-forward layers build predictions by promoting concepts in the vocabulary space. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 30–45, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Mor Geva, Yoav Goldberg, and Jonathan Berant. 2019. Are we modeling the task or the annotator? an investigation of annotator bias in natural language understanding datasets. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 1161–1166, Hong Kong, China. Association for Computational Linguistics.

Mor Geva, Roei Schuster, Jonathan Berant, and Omer Levy. 2021. Transformer feed-forward layers are key-value memories. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5484–5495, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Asma Ghandeharioun, Avi Caciularu, Adam Pearce, Lucas Dixon, and Mor Geva. 2024. Patchscopes: a unifying framework for inspecting hidden representations of language models. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org.

Gabriel Goh, Nick Cammarata †, Chelsea Voss †, Shan Carter, Michael Petrov, Ludwig Schubert, Alec Radford, and Chris Olah. 2021. Multimodal neurons in artificial neural networks. *Distill*. Https://distill.pub/2021/multimodal-neurons.

Noah D Goodman and Michael C Frank. 2016. Pragmatic language interpretation as probabilistic inference. *Trends in cognitive sciences*, 20(11):818–829.

Raghav Goyal, Samira Ebrahimi Kahou, Vincent Michalski, Joanna Materzynska, Susanne Westphal, Heuna Kim, Valentin Haenel, Ingo Fruend, Peter Yianilos, Moritz Mueller-Freitag, et al. 2017a. The" something something" video database for learning and evaluating visual common sense. In *Proceedings of the IEEE international conference on computer vision*, pages 5842–5850.

Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. 2017b. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6904–6913.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Klaus Greff, Francois Belletti, Lucas Beyer, Carl Doersch, Yilun Du, Daniel Duckworth, David J Fleet, Dan Gnanapragasam, Florian Golemo, Charles Herrmann, Thomas Kipf, Abhijit Kundu, Dmitry Lagun, Issam Laradji, Hsueh-Ti (Derek) Liu, Henning Meyer, Yishu Miao, Derek Nowrouzezahrai, Cengiz Oztireli, Etienne Pot, Noha Radwan, Daniel Rebain, Sara Sabour, Mehdi S. M. Sajjadi, Matan Sela, Vincent Sitzmann, Austin Stone, Deqing Sun, Suhani Vora, Ziyu Wang, Tianhao Wu, Kwang Moo Yi, Fangcheng Zhong, and Andrea Tagliasacchi. 2022. Kubric: a scalable dataset generator.

H Paul Grice. 1957. Meaning." the philosophical review 66: 377-88. 1969. *Utterer's Meaning and Intentions." The Philosophical Review*, 78:147–77.

Dirk Groeneveld, Iz Beltagy, Evan Walsh, Akshita Bhagia, Rodney Kinney, Oyvind Tafjord, Ananya Jha, Hamish Ivison, Ian Magnusson, Yizhong Wang, Shane Arora, David Atkinson, Russell Authur, Khyathi Chandu, Arman Cohan, Jennifer Dumas, Yanai Elazar, Yuling Gu, Jack Hessel, Tushar Khot, William Merrill, Jacob Morrison, Niklas Muennighoff, Aakanksha Naik, Crystal Nam, Matthew Peters, Valentina Pyatkin, Abhilasha Ravichander, Dustin Schwenk, Saurabh Shah, William Smith, Emma Strubell, Nishant Subramani, Mitchell Wortsman, Pradeep Dasigi, Nathan Lambert, Kyle Richardson, Luke Zettlemoyer, Jesse Dodge, Kyle Lo, Luca Soldaini, Noah Smith, and Hannaneh

Hajishirzi. 2024. OLMo: Accelerating the science of language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15789–15809, Bangkok, Thailand. Association for Computational Linguistics.

David Ha and Jürgen Schmidhuber. 2018. Recurrent world models facilitate policy evolution. In *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc.

Junlin Han, Shengbang Tong, David Fan, Yufan Ren, Koustuv Sinha, Philip Torr, and Filippos Kokkinos. 2025. Learning to see before seeing: Demystifying LLM visual priors from language pre-training. *arXiv preprint arXiv:2509.26625*.

Shibo Hao, Sainbayar Sukhbaatar, DiJia Su, Xian Li, Zhiting Hu, Jason E Weston, and Yuandong Tian. 2025. Training large language models to reason in a continuous latent space. In *Second Conference on Language Modeling*.

Stevan Harnad. 1990. The symbol grounding problem. *Physica D: Nonlinear Phenomena*, 42(1-3):335–346.

Will Douglas Heaven. 2021. This avocado armchair could be the future of AI. `https://www.technologyreview.com/2021/01/05/1015754/avocado-armchair-future-ai-openai-deep-learning-nlp-gpt3-computer-vision-commo` MIT Technology Review.

Irene Heim and Angelika Kratzer. 1998. *Semantics in Generative Grammar*. Number 13 in Blackwell Textbooks in Linguistics. Blackwell, Malden, MA.

Lisa Anne Hendricks and Aida Nematzadeh. 2021a. Probing Image-Language Transformers for Verb Understanding. *arXiv:2106.09141 [cs]*. ArXiv: 2106.09141.

Lisa Anne Hendricks and Aida Nematzadeh. 2021b. Probing image-language transformers for verb understanding. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3635–3644, Online. Association for Computational Linguistics.

Paloma García-de Herreros, Vagrant Gautam, Philipp Slusallek, Dietrich Klakow, and Marius Mosbach. 2024. What explains the success of cross-modal fine-tuning with ORCA? In *Proceedings of the Fifth Workshop on Insights from*

Negative Results in NLP, pages 8–16, Mexico City, Mexico. Association for Computational Linguistics.

Amir Hertz, Ron Mokady, Jay Tenenbaum, Kfir Aberman, Yael Pritch, and Daniel Cohen-Or. 2022. Prompt-to-prompt image editing with cross attention control.

Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Joseph Le Bras, and Yejin Choi. 2021. Clipscore: A reference-free evaluation metric for image captioning. *ArXiv*, abs/2104.08718.

Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. 2017. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Felix Hill. 2022. Noncompositional. `https://fh295.github.io/noncompositional.html`. Blog post.

Tobias Hinz, Stefan Heinrich, and Stefan Wermter. 2020. Semantic object accuracy for generative text-to-image synthesis. *IEEE transactions on pattern analysis and machine intelligence*.

Seunghoon Hong, Dingdong Yang, Jongwook Choi, and Honglak Lee. 2018. Inferring semantic layout for hierarchical text-to-image synthesis. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7986–7994.

Matthew Honnibal and Ines Montani. 2017. spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. To appear.

Matthew Honnibal, Ines Montani, Sofie Van Landeghem, and Adriane Boyd. 2020. spaCy: Industrial-strength Natural Language Processing in Python.

Arian Hosseini, Siva Reddy, Dzmitry Bahdanau, R Devon Hjelm, Alessandro Sordoni, and Aaron Courville. 2021. Understanding by understanding not: Modeling negation in language models. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1301–1312, Online. Association for Computational Linguistics.

Mehrdad Hosseinzadeh and Yang Wang. 2021. Image change captioning by learning from an auxiliary task. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2725–2734.

Cheng-Yu Hsieh, Jieyu Zhang, Zixian Ma, Aniruddha Kembhavi, and Ranjay Krishna. 2024. Sugarcrepe: Fixing hackable benchmarks for vision-language compositionality. *Advances in neural information processing systems*, 36.

Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Hexiang Hu, Ishan Misra, and Laurens van der Maaten. 2019. Binary Image Selection (BISON): Interpretable Evaluation of Visual Grounding. *arXiv preprint arXiv:1901.06595*.

Yushi Hu, Benlin Liu, Jungo Kasai, Yizhong Wang, Mari Ostendorf, Ranjay Krishna, and Noah A Smith. 2023. Tifa: Accurate and interpretable text-to-image faithfulness evaluation with question answering. *arXiv preprint arXiv:2303.11897*.

Yi Huang, Jiancheng Huang, Yifan Liu, Mingfu Yan, Jiaxi Lv, Jianzhuang Liu, Wei Xiong, He Zhang, Shifeng Chen, and Liangliang Cao. 2024. Diffusion model-based image editing: A survey. *arXiv preprint arXiv:2402.17525*.

Drew A Hudson and Christopher D Manning. 2019. Gqa: A new dataset for real-world visual reasoning and compositional question answering. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 6700–6709.

Minyoung Huh, Brian Cheung, Tongzhou Wang, and Phillip Isola. 2024. Position: The platonic representation hypothesis. In *Forty-first International Conference on Machine Learning*.

Mude Hui, Siwei Yang, Bingchen Zhao, Yichun Shi, Heng Wang, Peng Wang, Yuyin Zhou, and Cihang Xie. 2024. Hq-edit: A high-quality dataset for instruction-based image editing. *arXiv preprint arXiv:2404.09990*.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, et al. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.

brian ichter, Anthony Brohan, Yevgen Chebotar, Chelsea Finn, Karol Hausman, Alexander Herzog, Daniel Ho, Julian Ibarz, Alex Irpan, Eric Jang, Ryan Julian, Dmitry Kalashnikov, Sergey Levine, Yao Lu, Carolina Parada, Kanishka Rao, Pierre Sermanet, Alexander T Toshev, Vincent Vanhoucke, Fei Xia, Ted Xiao, Peng Xu, Mengyuan Yan, Noah Brown, Michael Ahn, Omar Cortes, Nicolas Sievers, Clayton Tan, Sichun Xu, Diego Reyes, Jarek Rettinghouse, Jornell Quiambao, Peter Pastor, Linda Luu, Kuang-Huei Lee, Yuheng Kuang, Sally Jesmonth, Nikhil J. Joshi, Kyle Jeffrey, Rosario Jauregui Ruano, Jasmine Hsu, Keerthana Gopalakrishnan, Byron David, Andy Zeng, and Chuyuan Kelly Fu. 2023. Do as i can, not as i say: Grounding language in robotic affordances. In *Proceedings of The 6th Conference on Robot Learning*, volume 205 of *Proceedings of Machine Learning Research*, pages 287–318. PMLR.

Priyank Jaini, Kevin Clark, and Robert Geirhos. 2024. Intriguing properties of generative classifiers. In *The Twelfth International Conference on Learning Representations*.

Sepehr Janghorbani and Gerard De Melo. 2023. Multi-modal bias: Introducing a framework for stereotypical bias assessment beyond gender and race in vision–language models. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 1717–1727, Dubrovnik, Croatia. Association for Computational Linguistics.

Harsh Jhamtani and Taylor Berg-Kirkpatrick. 2018a. Learning to Describe Differences Between Pairs of Similar Images. *arXiv:1808.10584 [cs]*. ArXiv: 1808.10584.

Harsh Jhamtani and Taylor Berg-Kirkpatrick. 2018b. Learning to describe differences between pairs of similar images. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 4024–4034, Brussels, Belgium. Association for Computational Linguistics.

Jingwei Ji, Ranjay Krishna, Li Fei-Fei, and Juan Carlos Niebles. 2020. Action genome: Actions as compositions of spatio-temporal scene graphs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10236–10247.

Chaoya Jiang, Haiyang Xu, Mengfan Dong, Jiaxing Chen, Wei Ye, Ming Yan, Qinghao Ye, Ji Zhang, Fei Huang, and Shikun Zhang. 2024a. Hallucination augmented contrastive learning for multimodal large language model. In

Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), pages 27036–27046.

Dongfu Jiang, Xuan He, Huaye Zeng, Cong Wei, Max Ku, Qian Liu, and Wenhu Chen. 2024b. Mantis: Interleaved multi-image instruction tuning. *Transactions on Machine Learning Research*.

Jiachen Jiang, Jinxin Zhou, Bo Peng, Xia Ning, and Zhihui Zhu. 2025a. Analyzing fine-grained alignment and enhancing vision understanding in multimodal language models. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

Nicholas Jiang, Anish Kachinthaya, Suzanne Petryk, and Yossi Gandelsman. 2025b. Interpreting and editing vision-language representations to mitigate hallucinations. In *The Thirteenth International Conference on Learning Representations*.

Ziyan Jiang, Rui Meng, Xinyi Yang, Semih Yavuz, Yingbo Zhou, and Wenhu Chen. 2025c. VLM2vec: Training vision-language models for massive multimodal embedding tasks. In *The Thirteenth International Conference on Learning Representations*.

Sonia Joseph, Quentin Garrido, Randall Balestriero, Matthew Kowal, Thomas Fel, Shahab Bakhtiari, Blake Richards, and Mike Rabbat. 2026. Interpreting physics in video world models. In *Forty-third International Conference on Machine Learning*.

Amita Kamath, Jack Hessel, and Kai-Wei Chang. 2023. What's "up" with vision-language models? investigating their struggle with spatial reasoning. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9161–9175, Singapore. Association for Computational Linguistics.

Bingyi Kang, Yang Yue, Rui Lu, Zhijie Lin, Yang Zhao, Kaixin Wang, Gao Huang, and Jiashi Feng. 2025. How far is video generation from world model: A physical law perspective. In *Forty-second International Conference on Machine Learning*.

Nora Kassner and Hinrich Schütze. 2020. Negated and misprimed probes for pretrained language models: Birds can talk, but cannot fly. In *Proceedings*

of the 58th Annual Meeting of the Association for Computational Linguistics, pages 7811–7818, Online. Association for Computational Linguistics.

Sahar Kazemzadeh, Vicente Ordonez, Mark Matten, and Tamara Berg. 2014. Referitgame: Referring to objects in photographs of natural scenes. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 787–798.

Jin-Hwa Kim, Yunji Kim, Jiyoung Lee, Kang Min Yoo, and Sang-Woo Lee. 2022. Mutual information divergence: A unified metric for multimodal generative models. In *Advances in Neural Information Processing Systems*.

Minji Kim, Taekyung Kim, and Bohyung Han. 2026. Map the flow: Revealing hidden pathways of information in VideoLLMs. In *International Conference on Learning Representations*.

Moo Jin Kim, Karl Pertsch, Siddharth Karamcheti, Ted Xiao, Ashwin Balakrishna, Suraj Nair, Rafael Rafailov, Ethan P Foster, Pannag R Sanketi, Quan Vuong, Thomas Kollar, Benjamin Burchfiel, Russ Tedrake, Dorsa Sadigh, Sergey Levine, Percy Liang, and Chelsea Finn. 2024. OpenVLA: An open-source vision-language-action model. In *8th Annual Conference on Robot Learning*.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. 2023. Segment anything. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4015–4026.

Jing Yu Koh, Robert Lo, Lawrence Jang, Vikram Duvvur, Ming Lim, Po-Yu Huang, Graham Neubig, Shuyan Zhou, Russ Salakhutdinov, and Daniel Fried. 2024. VisualWebArena: Evaluating multimodal agents on realistic visual web tasks. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 881–905, Bangkok, Thailand. Association for Computational Linguistics.

Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A. Shamma, Michael S. Bernstein, and Li Fei-Fei. 2017. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision*, 123(1):32–73.

Aravind Krishnan, Siva Reddy, and Marius Mosbach. 2025. Not all data are unlearned equally. In *Second Conference on Language Modeling*.

Benno Krojer, Vaibhav Adlakha, Vibhav Vineet, Yash Goyal, Edoardo Ponti, and Siva Reddy. 2022. Image retrieval from contextual descriptions. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3426–3440, Dublin, Ireland. Association for Computational Linguistics.

Benno Krojer, Mojtaba Komeili, Candace Ross, Quentin Garrido, Koustuv Sinha, Nicolas Ballas, and Mido Assran. 2025. A shortcut-aware video-QA benchmark for physical understanding via minimal video pairs. *Transactions on Machine Learning Research*.

Benno Krojer, Shravan Nayak, Oscar Mañas, Vaibhav Adlakha, Desmond Elliott, Siva Reddy, and Marius Mosbach. 2026. Latentlens: Revealing highly interpretable visual tokens in llms. In *Proceedings of the 43rd International Conference on Machine Learning (ICML)*.

Benno Krojer, Elinor Poole-Dayan, Vikram Voleti, Chris Pal, and Siva Reddy. 2023a. Are diffusion models vision-and-language reasoners? In *NeurIPS*.

Benno Krojer, Elinor Poole-Dayan, Vikram Voleti, Christopher Pal, and Siva Reddy. 2023b. Are diffusion models vision-and-language reasoners? In *Thirty-seventh Conference on Neural Information Processing Systems*.

Benno Krojer, Dheeraj Vattikonda, Luis Lara, Varun Jampani, Eva Portelance, Christopher Pal, and Siva Reddy. 2024a. Learning action and reasoning-centric image editing from videos and simulation. In *The Thirty-eight Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Benno Krojer, Dheeraj Vattikonda, Luis Lara, Varun Jampani, Eva Portelance, Christopher Pal, and Siva Reddy. 2024b. Learning action and reasoning-centric image editing from videos and simulation. In *Advances in Neural Information Processing Systems*, volume 37, pages 38035–38078. Curran Associates, Inc.

Max Ku, Dongfu Jiang, Cong Wei, Xiang Yue, and Wenhu Chen. 2024. VIEScore: Towards explainable metrics for conditional image synthesis evaluation. In

Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 12268–12290, Bangkok, Thailand. Association for Computational Linguistics.

Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, Tom Duerig, and Vittorio Ferrari. 2020. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *IJCV*.

George Lakoff and Mark Johnson. 1980. *Metaphors We Live By*. University of Chicago Press, Chicago.

Angeliki Lazaridou, Karl Moritz Hermann, Karl Tuyls, and Stephen Clark. 2018. Emergence of linguistic communication from referential games with symbolic and pixel input. In *International Conference on Learning Representations*.

Angeliki Lazaridou, Alexander Peysakhovich, and Marco Baroni. 2017. Multi-agent cooperation and the emergence of (natural) language. In *International Conference on Learning Representations*.

Jie Lei, Tamara Berg, and Mohit Bansal. 2023. Revealing single frame bias for video-and-language learning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 487–507, Toronto, Canada. Association for Computational Linguistics.

Brian Lester, Rami Al-Rfou, and Noah Constant. 2021. The power of scale for parameter-efficient prompt tuning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3045–3059, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Hector Levesque, Ernest Davis, and Leora Morgenstern. 2012. The winograd schema challenge. In *Thirteenth international conference on the principles of knowledge representation and reasoning*.

Martha Lewis, Qinan Yu, Jack Merullo, and Ellie Pavlick. 2022. Does clip bind concepts? probing compositionality in large image models. *arXiv preprint arXiv:2212.10537*.

Alexander C Li, Mihir Prabhudesai, Shivam Duggal, Ellis Brown, and Deepak Pathak. 2023a. Your diffusion model is secretly a zero-shot classifier. *arXiv preprint arXiv:2303.16203*.

Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. 2024a. Llava-onevision: Easy visual task transfer. *arXiv preprint arXiv:2408.03326*.

Bo Li, Yuanhan Zhang, Dong Guo, Renrui Zhang, Feng Li, Hao Zhang, Kaichen Zhang, Peiyuan Zhang, Yanwei Li, Ziwei Liu, and Chunyuan Li. 2025a. LLaVA-onevision: Easy visual task transfer. *Transactions on Machine Learning Research*.

Chengzu Li, Wenshan Wu, Huanyu Zhang, Yan Xia, Shaoguang Mao, Li Dong, Ivan Vulić, and Furu Wei. 2025b. Imagine while reasoning in space: Multimodal visualization-of-thought. *arXiv preprint arXiv:2501.07542*.

Jiaang Li, Yova Kementchedjhieva, Constanza Fierro, and Anders Søgaard. 2024b. Do vision and language models share concepts? a vector space alignment study. *Transactions of the Association for Computational Linguistics*, 12:1232–1249.

Junnan Li, Dongxu Li, Silvio Savarese, and Steven Hoi. 2023b. Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models. In *Proceedings of the 40th International Conference on Machine Learning*.

Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. 2022a. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *International Conference on Machine Learning*, pages 12888–12900. PMLR.

Junnan Li, Yongkang Wong, Qi Zhao, and Mohan S Kankanhalli. 2019. Video storytelling: Textual summaries for events. *IEEE Transactions on Multimedia*, 22(2):554–565.

Zhiheng Li, Martin Renqiang Min, Kai Li, and Chenliang Xu. 2022b. Stylet2i: Toward compositional and high-fidelity text-to-image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18197–18207.

Victor Weixin Liang, Yuhui Zhang, Yongchan Kwon, Serena Yeung, and James Y Zou. 2022. Mind the gap: Understanding the modality gap in multi-modal contrastive representation learning. *Advances in Neural Information Processing Systems*, 35:17612–17625.

Weixin Liang, LILI YU, Liang Luo, Srini Iyer, Ning Dong, Chunting Zhou, Gargi Ghosh, Mike Lewis, Wen tau Yih, Luke Zettlemoyer, and Xi Victoria Lin. 2025. Mixture-of-transformers: A sparse and scalable architecture for multi-modal foundation models. *Transactions on Machine Learning Research*.

Jiaqi Liao, Yuwei Niu, Fanqing Meng, Hao Li, Changyao Tian, Yinuo Du, Yuwen Xiong, Dianqi Li, Xizhou Zhu, Li Yuan, Jifeng Dai, and Yu Cheng. 2025. Langbridge: Interpreting image as a combination of language embeddings. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 23752–23762.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer.

Zachary C. Lipton. 2018. The mythos of model interpretability. *Queue*, 16(3):31–57.

Fangyu Liu, Emanuele Bugliarello, Edoardo Maria Ponti, Siva Reddy, Nigel Collier, and Desmond Elliott. 2021. Visually grounded reasoning across languages and cultures. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 10467–10485, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Haotian Liu, Chunyuan Li, Yuheng Li, and Yong Jae Lee. 2024a. Improved baselines with visual instruction tuning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26296–26306.

Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. 2024b. LLaVA-NeXT: Improved reasoning, ocr, and world knowledge. `https://llava-vl.github.io/blog/2024-01-30-llava-next/`.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023. Visual instruction tuning. *Advances in neural information processing systems*, 36:34892–34916.

Tania Lombrozo. 2006. The structure and function of explanations. *Trends in cognitive sciences*, 10(10):464–470.

Jiasen Lu, Dhruv Batra, Devi Parikh, and Stefan Lee. 2019. ViLBERT: Pretraining Task-Agnostic Visiolinguistic Representations for Vision-and-Language Tasks. *Advances in Neural Information Processing Systems*, 32.

Kevin Lu, Aditya Grover, Pieter Abbeel, and Igor Mordatch. 2022. Frozen pretrained transformers as universal computation engines. In *Proceedings of the AAAI conference on artificial intelligence*, volume 36, pages 7628–7636.

Alexandra Sasha Luccioni, Christopher Akiki, Margaret Mitchell, and Yacine Jernite. 2023. Stable bias: Analyzing societal representations in diffusion models.

Shweta Mahajan, Shreya Kadambi, Hoang Le, Munawar Hayat, and Fatih Porikli. 2025. Do-undo: Generating and reversing physical actions in vision-language models. *arXiv preprint arXiv:2512.13609*.

Oscar Mañas, Pau Rodriguez Lopez, Saba Ahmadi, Aida Nematzadeh, Yash Goyal, and Aishwarya Agrawal. 2023. Mapl: Parameter-efficient adaptation of unimodal pre-trained models for vision-language few-shot prompting. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 2523–2548.

Junhua Mao, Jonathan Huang, Alexander Toshev, Oana Camburu, Alan Yuille, and Kevin Murphy. 2016. Generation and comprehension of unambiguous object descriptions. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11–20.

Arnau Marin-Llobet, Simon Henniger, and Mahzarin R. Banaji. 2026. Vision-language models suppress female representations under ambiguous input.

Ahmed Masry, Juan A. Rodriguez, Tianyu Zhang, Suyuchen Wang, Chao Wang, Aarash Feizi, Akshay Kalkunte Suresh, Abhay Puri, Xiangru Jian, Pierre-Andre Noel, Sathwik Tejaswi Madhusudhan, Marco Pedersoli, Bang Liu, Nicolas Chapados, Yoshua Bengio, Enamul Hoque, Christopher Pal, Issam H. Laradji, David Vazquez, Perouz Taslakian, Spandana Gella, and Sai Rajeswar. 2025. Alignvlm: Bridging vision and language latent spaces for multimodal document understanding. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

R. Thomas McCoy, Ellie Pavlick, and Tal Linzen. 2019. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3428–3448, Florence, Italy. Association for Computational Linguistics.

Nicholas Meade, Elinor Poole-Dayan, and Siva Reddy. 2022. An empirical survey of the effectiveness of debiasing techniques for pre-trained language models. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1878–1898, Dublin, Ireland. Association for Computational Linguistics.

Jack Merullo, Louis Castricato, Carsten Eickhoff, and Ellie Pavlick. 2023. Linearly mapping from image to text space. In *The Eleventh International Conference on Learning Representations*.

Oscar Michel, Anand Bhattad, Eli VanderBilt, Ranjay Krishna, Aniruddha Kembhavi, and Tanmay Gupta. 2023. Object 3dit: Language-guided 3d-aware image editing. *ArXiv*, abs/2307.11073.

Antoine Miech, Jean-Baptiste Alayrac, Ivan Laptev, Josef Sivic, and Andrew Zisserman. 2021. Thinking Fast and Slow: Efficient Text-to-Visual Retrieval with Transformers. ArXiv:2103.16553 [cs].

Antoine Miech, Dimitri Zhukov, Jean-Baptiste Alayrac, Makarand Tapaswi, Ivan Laptev, and Josef Sivic. 2019. Howto100m: Learning a text-video embedding by watching hundred million narrated video clips. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 2630–2640.

Anish Mittal, Anush Krishna Moorthy, and Alan Conrad Bovik. 2012. No-reference image quality assessment in the spatial domain. *IEEE Transactions on image processing*, 21(12):4695–4708.

Ron Mokady, Amir Hertz, and Amit H Bermano. 2021. Clipcap: Clip prefix for image captioning. *arXiv preprint arXiv:2111.09734*.

Moin Nadeem, Anna Bethke, and Siva Reddy. 2021. StereoSet: Measuring stereotypical bias in pretrained language models. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 5356–5371, Online. Association for Computational Linguistics.

Clement Neo, Luke Ong, Philip Torr, Mor Geva, David Krueger, and Fazl Barez. 2025. Towards interpreting visual information processing in vision-language models. In *The Thirteenth International Conference on Learning Representations*.

Yaniv Nikankin, Dana Arad, Yossi Gandelsman, and Yonatan Belinkov. 2025. Same task, different circuits: Disentangling modality-specific mechanisms in VLMs. In *Advances in Neural Information Processing Systems*.

Yulei Niu, Wenliang Guo, Long Chen, Xudong Lin, and Shih-Fu Chang. 2024. SCHEMA: State CHanges MAtter for procedure planning in instructional videos. In *The Twelfth International Conference on Learning Representations*.

nostalgebraist. 2020. interpreting gpt: the logit lens.

Tolúlopé Ògúnrèmí, Christopher D Manning, Dan Jurafsky, and Karen Livescu. 2025. Transcribe, translate, or transliterate: An investigation of intermediate representations in spoken language models. *arXiv preprint arXiv:2510.02569*.

Dan Oneata, Desmond Elliott, and Stella Frank. 2025. Seeing what tastes good: Revisiting multimodal distributional semantics in the billion parameter era. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 24174–24191, Vienna, Austria. Association for Computational Linguistics.

OpenAI. 2024. GPT-4 technical report.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. 2023. Dinov2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*.

Jiefu Ou, Benno Krojer, and Daniel Fried. 2023. Pragmatic inference with a CLIP listener for contrastive captioning. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 1904–1917, Toronto, Canada. Association for Computational Linguistics.

Alexander Pan, Lijie Chen, and Jacob Steinhardt. 2024. Latentqa: Teaching llms to decode activations into natural language. *arXiv preprint arXiv:2412.08686*.

Isabel Papadimitriou, Huangyuan Su, Thomas Fel, Sham Kakade, and Stephanie Gil. 2025. Interpreting the linear structure of vision-language model embedding spaces. *arXiv preprint arXiv:2504.11695*.

Letitia Parcalabescu, Michele Cafagna, Lilitta Muradjan, Anette Frank, Iacer Calixto, and Albert Gatt. 2022. VALSE: A task-independent benchmark for vision and language models centered on linguistic phenomena. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8253–8280, Dublin, Ireland. Association for Computational Linguistics.

Dong Huk Park, Trevor Darrell, and Anna Rohrbach. 2019. Robust change captioning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4624–4633.

Seongheon Park and Sharon Li. 2025. GLSim: Detecting object hallucinations in LVLMs via global-local similarity. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

Omkar M. Parkhi, Andrea Vedaldi, Andrew Zisserman, and C. V. Jawahar. 2012. Cats and dogs. *IEEE Conference on Computer Vision and Pattern Recognition*.

Roma Patel and Ellie Pavlick. 2022. Mapping language models to grounded conceptual spaces. In *International conference on learning representations*.

Karalyn E Patterson, Peter J. Nestor, and Timothy T. Rogers. 2007. Where do you know what you know? the representation of semantic knowledge in the human brain. *Nature Reviews Neuroscience*, 8:976–987.

Diane Pecher and Rolf A Zwaan. 2005. *Grounding cognition: The role of perception and action in memory, language, and thinking*. Cambridge University Press.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 2227–2237, New Orleans, Louisiana. Association for Computational Linguistics.

Anirudh Phukan, Divyansh Divyansh, Harshit Kumar Morj, Vaishnavi Vaishnavi, Apoorv Saxena, and Koustava Goswami. 2025. Beyond logit lens: Contextual embeddings for robust hallucination detection & grounding in vlms. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 9661–9675.

Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. 2024. SDXL: Improving latent diffusion models for high-resolution image synthesis. In *The Twelfth International Conference on Learning Representations*.

Ben Poole, Ajay Jain, Jonathan T Barron, and Ben Mildenhall. 2022. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv preprint arXiv:2209.14988*.

Yulu Qin, Dheeraj Varghese, Adam Dahlgren Lindström, Lucia Donatelli, Kanishka Misra, and Najoung Kim. 2026. Vision-and-language training helps deploy taxonomic knowledge but does not fundamentally alter it. *Advances in Neural Information Processing Systems*, 38:12273–12301.

Leigang Qu, Wenjie Wang, Yongqi Li, Hanwang Zhang, Liqiang Nie, and Tat-Seng Chua. 2024. Discriminative probing and tuning for text-to-image generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7434–7444.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021a. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763. PMLR.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021b. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763. PMLR.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. 2021c. Learning transferable visual models from natural language supervision. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763. PMLR.

Daking Rai, Yilun Zhou, Shi Feng, Abulhair Saparov, and Ziyu Yao. 2024. A practical review of mechanistic interpretability for transformer-based language models. *arXiv preprint arXiv:2407.02646*.

Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. 2022. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 1(2):3.

Arijit Ray, Ahmed Abdelkader, Chengzhi Mao, Bryan A Plummer, Kate Saenko, Ranjay Krishna, Leonidas Guibas, and Wen-Sheng Chu. 2026. Mull-tokens: Modality-agnostic latent thinking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9477–9488.

Ronan Riochet, Mario Ynocente Castro, Mathieu Bernard, Adam Lerer, Rob Fergus, Véronique Izard, and Emmanuel Dupoux. 2022. Intphys 2019: A benchmark for visual intuitive physics understanding. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(9):5016–5025.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10684–10695.

E. Rosch, C. Mervis, Wayne D. Gray, David M. Johnson, and P. Boyes-Braem. 1976. Basic objects in natural categories. *Cognitive Psychology*, 8:382–439.

Candace Ross, Florian Bordes, Adina Williams, Polina Kirichenko, and Mark Ibrahim. 2026. What's in common? multimodal models hallucinate when reasoning across scenes. *Advances in Neural Information Processing Systems*, 38.

Noam Rotstein, Gal Yona, Daniel Silver, Roy Velich, David Bensaïd, and Ron Kimmel. 2025. Pathways on the image manifold: Image editing via video

generation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7857–7866.

Laura Ruis, Akbir Khan, Stella Biderman, Sara Hooker, Tim Rocktäschel, and Edward Grefenstette. 2023. The goldilocks of pragmatic understanding: Fine-tuning strategy matters for implicature resolution by llms. In *Advances in Neural Information Processing Systems*, volume 36, pages 20827–20905. Curran Associates, Inc.

Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L. Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, Seyedeh Sara Mahdavi, Raphael Gontijo Lopes, Tim Salimans, Jonathan Ho, David J. Fleet, and Mohammad Norouzi. 2022. Photorealistic text-to-image diffusion models with deep language understanding. *ArXiv*, abs/2205.11487.

Dvir Samuel, Rami Ben-Ari, Simon Raviv, Nir Darshan, and Gal Chechik. 2023. It is all about where you start: Text-to-image generation with seed selection. *arXiv preprint arXiv:2304.14530*.

Naomi Saphra and Sarah Wiegreffe. 2024. Mechanistic? In *Proceedings of the 7th BlackboxNLP Workshop*, pages 480–498, Miami, Florida, US. Association for Computational Linguistics.

Christoph Schuhmann, Richard Vencu, Romain Beaumont, Robert Kaczmarczyk, Clayton Mullis, Aarush Katta, Theo Coombes, Jenia Jitsev, and Aran Komatsuzaki. 2021. Laion-400m: Open dataset of clip-filtered 400 million image-text pairs. *arXiv preprint arXiv:2111.02114*.

Alessandro Pietro Serra, Francesco Ortu, Emanuele Panizon, Lucrezia Valeriani, Lorenzo Basile, Alessio Ansuini, Diego Doimo, and Alberto Cazzaniga. 2025. The narrow gate: Localized image-text communication in native multimodal models. In *Advances in Neural Information Processing Systems*.

Lee Sharkey, Bilal Chughtai, Joshua Batson, Jack Lindsey, Jeffrey Wu, Lucius Bushnaq, Nicholas Goldowsky-Dill, Stefan Heimersheim, Alejandro Ortega, Joseph Isaac Bloom, Stella Biderman, Adrià Garriga-Alonso, Arthur Conmy, Neel Nanda, Jessica Mary Rumbelow, Martin Wattenberg, Nandi Schoots, Joseph Miller, William Saunders, Eric J Michaud, Stephen Casper, Max Tegmark, David Bau, Eric Todd, Atticus Geiger, Mor Geva, Jesse Hoogland,

Daniel Murfet, and Thomas McGrath. 2025. Open problems in mechanistic interpretability. *Transactions on Machine Learning Research*.

Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. 2018. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2556–2565, Melbourne, Australia. Association for Computational Linguistics.

Junhong Shen, Liam Li, Lucio M. Dery, Corey Staten, Mikhail Khodak, Graham Neubig, and Ameet Talwalkar. 2023. Cross-modal fine-tuning: Align then refine. In *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 31030–31056. PMLR.

Shelly Sheynin, Adam Polyak, Uriel Singer, Yuval Kirstain, Amit Zohar, Oron Ashual, Devi Parikh, and Yaniv Taigman. 2023. Emu edit: Precise image editing via recognition and generation tasks. *arXiv preprint arXiv:2311.10089*.

Mustafa Shukor and Matthieu Cord. 2024. Implicit multimodal alignment: On the generalization of frozen LLMs to multimodal inputs. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Mustafa Shukor, Enrico Fini, Victor Guilherme Turrisi da Costa, Matthieu Cord, Joshua Susskind, and Alaaeldin El-Nouby. 2025. Scaling laws for native multimodal models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12–23.

Gunnar A Sigurdsson, Gül Varol, Xiaolong Wang, Ali Farhadi, Ivan Laptev, and Abhinav Gupta. 2016. Hollywood in homes: Crowdsourcing data collection for activity understanding. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14*, pages 510–526. Springer.

Ray Smith. 2007. An overview of the Tesseract OCR engine. In *Ninth international conference on document analysis and recognition (ICDAR 2007)*, volume 2, pages 629–633. IEEE.

Tomáš Souček, Dima Damen, Michael Wray, Ivan Laptev, and Josef Sivic. 2024. Genhowto: Learning to generate actions and state transformations from

instructional videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6561–6571.

Elizabeth S Spelke and Katherine D Kinzler. 2007. Core knowledge. *Developmental science*, 10(1):89–96.

Tejas Srinivasan and Yonatan Bisk. 2022. Worst of both worlds: Biases compound in pre-trained vision-and-language models. In *Proceedings of the 4th Workshop on Gender Bias in Natural Language Processing (GeBNLP)*, pages 77–85, Seattle, Washington. Association for Computational Linguistics.

Yongyi Su, Haojie Zhang, Shijie Li, Nanqing Liu, Jingyi Liao, Junyi Pan, Yuan Liu, Xiaofen Xing, Chong Sun, Chen Li, Nancy F. Chen, Shuicheng YAN, Xulei Yang, and Xun Xu. 2026. Patch-as-decodable-token: Towards unified multimodal vision tasks in MLLMs. In *The Fourteenth International Conference on Learning Representations*.

Zhaochen Su, Peng Xia, Hangyu Guo, Zhenhua Liu, Yan Ma, Xiaoye Qu, Jiaqi Liu, Yanshu Li, Kaide Zeng, Zhengyuan Yang, et al. 2025. Thinking with images for multimodal reasoning: Foundations, methods, and future frontiers. *arXiv preprint arXiv:2506.23918*.

Alane Suhr, Stephanie Zhou, Ally Zhang, Iris Zhang, Huajun Bai, and Yoav Artzi. 2019. A Corpus for Reasoning about Natural Language Grounded in Photographs. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6418–6428, Florence, Italy. Association for Computational Linguistics.

Hao Tan and Mohit Bansal. 2019. LXMERT: Learning cross-modality encoder representations from transformers. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5100–5111, Hong Kong, China. Association for Computational Linguistics.

Hao Tan, Franck Dernoncourt, Zhe L. Lin, Trung Bui, and Mohit Bansal. 2019. Expressing visual relationships via language. In *Annual Meeting of the Association for Computational Linguistics*.

Raphael Tang, Linqing Liu, Akshat Pandey, Zhiying Jiang, Gefei Yang, Karun Kumar, Pontus Stenetorp, Jimmy Lin, and Ferhan Ture. 2023. What the

DAAM: Interpreting stable diffusion using cross attention. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5644–5659, Toronto, Canada. Association for Computational Linguistics.

Chameleon Team. 2024. Chameleon: Mixed-modal early-fusion foundation models. *arXiv preprint arXiv:2405.09818*.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M. Dai, Anja Hauth, Katie Millican, David Silver, Melvin Johnson, Ioannis Antonoglou, Julian Schrittwieser, Amelia Glaese, Jilin Chen, Emily Pitler, Timothy Lillicrap, Angeliki Lazaridou, Orhan Firat, James Molloy, Michael Isard, Paul R. Barham, Tom Hennigan, Benjamin Lee, Fabio Viola, Malcolm Reynolds, Yuanzhong Xu, Ryan Doherty, Eli Collins, Clemens Meyer, Eliza Rutherford, Erica Moreira, Kareem Ayoub, Megha Goel, Jack Krawczyk, Cosmo Du, Ed Chi, Heng-Tze Cheng, Eric Ni, Purvi Shah, Patrick Kane, Betty Chan, Manaal Faruqui, Aliaksei Severyn, Hanzhao Lin, YaGuang Li, Yong Cheng, Abe Ittycheriah, Mahdis Mahdieh, Mia Chen, Pei Sun, Dustin Tran, Sumit Bagri, Balaji Lakshminarayanan, Jeremiah Liu, Andras Orban, Fabian Güra, Hao Zhou, Xinying Song, Aurelien Boffy, Harish Ganapathy, Steven Zheng, HyunJeong Choe, Ágoston Weisz, Tao Zhu, Yifeng Lu, Siddharth Gopal, Jarrod Kahn, Maciej Kula, Jeff Pitman, Rushin Shah, Emanuel Taropa, Majd Al Merey, Martin Baeuml, Zhifeng Chen, Laurent El Shafey, Yujing Zhang, Olcan Sercinoglu, George Tucker, Enrique Piqueras, Maxim Krikun, Iain Barr, Nikolay Savinov, Ivo Danihelka, Becca Roelofs, Anaïs White, Anders Andreassen, Tamara von Glehn, Lakshman Yagati, Mehran Kazemi, Lucas Gonzalez, Misha Khalman, Jakub Sygnowski, Alexandre Frechette, Charlotte Smith, Laura Culp, Lev Proleev, Yi Luan, Xi Chen, James Lottes, Nathan Schucher, Federico Lebron, Alban Rrustemi, Natalie Clay, Phil Crone, Tomas Kocisky, Jeffrey Zhao, Bartek Perz, Dian Yu, Heidi Howard, Adam Bloniarz, Jack W. Rae, Han Lu, Laurent Sifre, Marcello Maggioni, Fred Alcober, Dan Garrette, Megan Barnes, Shantanu Thakoor, Jacob Austin, Gabriel Barth-Maron, William Wong, Rishabh Joshi, Rahma Chaabouni, Deeni Fatiha, Arun Ahuja, Gaurav Singh Tomar, Evan Senter, Martin Chadwick, Ilya Kornakov, Nithya Attaluri, Iñaki Iturrate, Ruibo Liu, Yunxuan Li, Sarah Cogan, Jeremy Chen, Chao Jia, Chenjie Gu, Qiao Zhang, Jordan Grimstad, Ale Jakse Hartman, Xavier Garcia, Thanumalayan Sankaranarayana Pillai, Jacob Devlin, Michael Laskin, Diego de Las Casas, Dasha

Valter, Connie Tao, Lorenzo Blanco, Adrià Puigdomènech Badia, David Reitter, Mianna Chen, Jenny Brennan, Clara Rivera, Sergey Brin, Shariq Iqbal, Gabriela Surita, Jane Labanowski, Abhi Rao, Stephanie Winkler, Emilio Parisotto, Yiming Gu, Kate Olszewska, Ravi Addanki, Antoine Miech, Annie Louis, Denis Teplyashin, Geoff Brown, Elliot Catt, Jan Balaguer, Jackie Xiang, Pidong Wang, Zoe Ashwood, Anton Briukhov, Albert Webson, Sanjay Ganapathy, Smit Sanghavi, Ajay Kannan, Ming-Wei Chang, Axel Stjerngren, Josip Djolonga, Yuting Sun, Ankur Bapna, Matthew Aitchison, Pedram Pejman, Henryk Michalewski, Tianhe Yu, Cindy Wang, Juliette Love, Junwhan Ahn, Dawn Bloxwich, Kehang Han, Peter Humphreys, Thibault Sellam, James Bradbury, Varun Godbole, Sina Samangooei, Bogdan Damoc, Alex Kaskasoli, Sébastien M. R. Arnold, Vijay Vasudevan, Shubham Agrawal, Jason Riesa, Dmitry Lepikhin, Richard Tanburn, Srivatsan Srinivasan, Hyeontaek Lim, Sarah Hodkinson, Pranav Shyam, Johan Ferret, Steven Hand, Ankush Garg, Tom Le Paine, Jian Li, Yujia Li, Minh Giang, Alexander Neitz, Zaheer Abbas, Sarah York, Machel Reid, Elizabeth Cole, Aakanksha Chowdhery, Dipanjan Das, Dominika Rogozińska, Vitaliy Nikolaev, Pablo Sprechmann, Zachary Nado, Lukas Zilka, Flavien Prost, Luheng He, Marianne Monteiro, Gaurav Mishra, Chris Welty, Josh Newlan, Dawei Jia, Miltiadis Allamanis, Clara Huiyi Hu, Raoul de Liedekerke, Justin Gilmer, Carl Saroufim, Shruti Rijhwani, Shaobo Hou, Disha Shrivastava, Anirudh Baddepudi, Alex Goldin, Adnan Ozturel, Albin Cassirer, Yunhan Xu, Daniel Sohn, Devendra Sachan, Reinald Kim Amplayo, Craig Swanson, Dessie Petrova, Shashi Narayan, Arthur Guez, Siddhartha Brahma, Jessica Landon, Miteyan Patel, Ruizhe Zhao, Kevin Villela, Luyu Wang, Wenhao Jia, Matthew Rahtz, Mai Giménez, Legg Yeung, James Keeling, Petko Georgiev, Diana Mincu, Boxi Wu, Salem Haykal, Rachel Saputro, Kiran Vodrahalli, James Qin, Zeynep Cankara, Abhanshu Sharma, Nick Fernando, Will Hawkins, Behnam Neyshabur, Solomon Kim, Adrian Hutter, Priyanka Agrawal, Alex Castro-Ros, George van den Driessche, Tao Wang, Fan Yang, Shuo yiin Chang, Paul Komarek, Ross McIlroy, Mario Lučić, Guodong Zhang, Wael Farhan, Michael Sharman, Paul Natsev, Paul Michel, Yamini Bansal, Siyuan Qiao, Kris Cao, Siamak Shakeri, Christina Butterfield, Justin Chung, Paul Kishan Rubenstein, Shivani Agrawal, Arthur Mensch, Kedar Soparkar, Karel Lenc, Timothy Chung, Aedan Pope, Loren Maggiore, Jackie Kay, Priya Jhakra, Shibo Wang, Joshua Maynez, Mary Phuong, Taylor Tobin, Andrea Tacchetti, Maja Trebacz, Kevin Robinson, Yash Katariya, Sebastian Riedel, Paige Bailey, Kefan Xiao, Nimesh Ghelani, Lora Aroyo, Ambrose Slone, Neil Houlsby, Xuehan Xiong, Zhen Yang, Elena Gribovskaya,

Jonas Adler, Mateo Wirth, Lisa Lee, Music Li, Thais Kagohara, Jay Pavagadhi, Sophie Bridgers, Anna Bortsova, Sanjay Ghemawat, Zafarali Ahmed, Tianqi Liu, Richard Powell, Vijay Bolina, Mariko Iinuma, Polina Zablotskaia, James Besley, Da-Woon Chung, Timothy Dozat, Ramona Comanescu, Xiance Si, Jeremy Greer, Guolong Su, Martin Polacek, Raphaël Lopez Kaufman, Simon Tokumine, Hexiang Hu, Elena Buchatskaya, Yingjie Miao, Mohamed Elhawaty, Aditya Siddhant, Nenad Tomasev, Jinwei Xing, Christina Greer, Helen Miller, Shereen Ashraf, Aurko Roy, Zizhao Zhang, Ada Ma, Angelos Filos, Milos Besta, Rory Blevins, Ted Klimenko, Chih-Kuan Yeh, Soravit Changpinyo, Jiaqi Mu, Oscar Chang, Mantas Pajarskas, Carrie Muir, Vered Cohen, Charline Le Lan, Krishna Haridasan, Amit Marathe, Steven Hansen, Sholto Douglas, Rajkumar Samuel, Mingqiu Wang, Sophia Austin, Chang Lan, Jiepu Jiang, Justin Chiu, Jaime Alonso Lorenzo, Lars Lowe Sjösund, Sébastien Cevey, Zach Gleicher, Thi Avrahami, Anudhyan Boral, Hansa Srinivasan, Vittorio Selo, Rhys May, Konstantinos Aisopos, Léonard Hussenot, Livio Baldini Soares, Kate Baumli, Michael B. Chang, Adrià Recasens, Ben Caine, Alexander Pritzel, Filip Pavetic, Fabio Pardo, Anita Gergely, Justin Frye, Vinay Ramasesh, Dan Horgan, Kartikeya Badola, Nora Kassner, Subhrajit Roy, Ethan Dyer, Víctor Campos Campos, Alex Tomala, Yunhao Tang, Dalia El Badawy, Elspeth White, Basil Mustafa, Oran Lang, Abhishek Jindal, Sharad Vikram, Zhitao Gong, Sergi Caelles, Ross Hemsley, Gregory Thornton, Fangxiaoyu Feng, Wojciech Stokowiec, Ce Zheng, Phoebe Thacker, Çağlar Ünlü, Zhishuai Zhang, Mohammad Saleh, James Svensson, Max Bileschi, Piyush Patil, Ankesh Anand, Roman Ring, Katerina Tsihlas, Arpi Vezer, Marco Selvi, Toby Shevlane, Mikel Rodriguez, Tom Kwiatkowski, Samira Daruki, Keran Rong, Allan Dafoe, Nicholas FitzGerald, Keren Gu-Lemberg, Mina Khan, Lisa Anne Hendricks, Marie Pellat, Vladimir Feinberg, James Cobon-Kerr, Tara Sainath, Maribeth Rauh, Sayed Hadi Hashemi, Richard Ives, Yana Hasson, Eric Noland, Yuan Cao, Nathan Byrd, Le Hou, Qingze Wang, Thibault Sottiaux, Michela Paganini, Jean-Baptiste Lespiau, Alexandre Moufarek, Samer Hassan, Kaushik Shivakumar, Joost van Amersfoort, Amol Mandhane, Pratik Joshi, Anirudh Goyal, Matthew Tung, Andrew Brock, Hannah Sheahan, Vedant Misra, Cheng Li, Nemanja Rakićević, Mostafa Dehghani, Fangyu Liu, Sid Mittal, Junhyuk Oh, Seb Noury, Eren Sezener, Fantine Huot, Matthew Lamm, Nicola De Cao, Charlie Chen, Sidharth Mudgal, Romina Stella, Kevin Brooks, Gautam Vasudevan, Chenxi Liu, Mainak Chain, Nivedita Melinkeri, Aaron Cohen, Venus Wang, Kristie Seymore, Sergey Zubkov, Rahul Goel, Summer Yue, Sai Krishnakumaran, Brian Albert, Nate Hurley, Mo-

toki Sano, Anhad Mohananey, Jonah Joughin, Egor Filonov, Tomasz Kępa, Yomna Eldawy, Jiawern Lim, Rahul Rishi, Shirin Badiezadegan, Taylor Bos, Jerry Chang, Sanil Jain, Sri Gayatri Sundara Padmanabhan, Subha Puttagunta, Kalpesh Krishna, Leslie Baker, Norbert Kalb, Vamsi Bedapudi, Adam Kurzrok, Shuntong Lei, Anthony Yu, Oren Litvin, Xiang Zhou, Zhichun Wu, Sam Sobell, Andrea Siciliano, Alan Papir, Robby Neale, Jonas Bragagnolo, Tej Toor, Tina Chen, Valentin Anklin, Feiran Wang, Richie Feng, Milad Gholami, Kevin Ling, Lijuan Liu, Jules Walter, Hamid Moghaddam, Arun Kishore, Jakub Adamek, Tyler Mercado, Jonathan Mallinson, Siddhinita Wandekar, Stephen Cagle, Eran Ofek, Guillermo Garrido, Clemens Lombriser, Maksim Mukha, Botu Sun, Hafeezul Rahman Mohammad, Josip Matak, Yadi Qian, Vikas Peswani, Pawel Janus, Quan Yuan, Leif Schelin, Oana David, Ankur Garg, Yifan He, Oleksii Duzhyi, Anton Älgmyr, Timothée Lottaz, Qi Li, Vikas Yadav, Luyao Xu, Alex Chinien, Rakesh Shivanna, Aleksandr Chuklin, Josie Li, Carrie Spadine, Travis Wolfe, Kareem Mohamed, Subhabrata Das, Zihang Dai, Kyle He, Daniel von Dincklage, Shyam Upadhyay, Akanksha Maurya, Luyan Chi, Sebastian Krause, Khalid Salama, Pam G Rabinovitch, Pavan Kumar Reddy M, Aarush Selvan, Mikhail Dektiarev, Golnaz Ghiasi, Erdem Guven, Himanshu Gupta, Boyi Liu, Deepak Sharma, Idan Heimlich Shtacher, Shachi Paul, Oscar Akerlund, François-Xavier Aubet, Terry Huang, Chen Zhu, Eric Zhu, Elico Teixeira, Matthew Fritze, Francesco Bertolini, Liana-Eleonora Marinescu, Martin Bölle, Dominik Paulus, Khyatti Gupta, Tejasi Latkar, Max Chang, Jason Sanders, Roopa Wilson, Xuewei Wu, Yi-Xuan Tan, Lam Nguyen Thiet, Tulsee Doshi, Sid Lall, Swaroop Mishra, Wanming Chen, Thang Luong, Seth Benjamin, Jasmine Lee, Ewa Andrejczuk, Dominik Rabiej, Vipul Ranjan, Krzysztof Styrc, Pengcheng Yin, Jon Simon, Malcolm Rose Harriott, Mudit Bansal, Alexei Robsky, Geoff Bacon, David Greene, Daniil Mirylenka, Chen Zhou, Obaid Sarvana, Abhimanyu Goyal, Samuel Andermatt, Patrick Siegler, Ben Horn, Assaf Israel, Francesco Pongetti, Chih-Wei "Louis" Chen, Marco Selvatici, Pedro Silva, Kathie Wang, Jackson Tolins, Kelvin Guu, Roey Yogev, Xiaochen Cai, Alessandro Agostini, Maulik Shah, Hung Nguyen, Noah Ó Donnaile, Sébastien Pereira, Linda Friso, Adam Stambler, Adam Kurzrok, Chenkai Kuang, Yan Romanikhin, Mark Geller, ZJ Yan, Kane Jang, Cheng-Chun Lee, Wojciech Fica, Eric Malmi, Qijun Tan, Dan Banica, Daniel Balle, Ryan Pham, Yanping Huang, Diana Avram, Hongzhi Shi, Jasjot Singh, Chris Hidey, Niharika Ahuja, Pranab Saxena, Dan Dooley, Srividya Pranavi Potharaju, Eileen O'Neill, Anand Gokulchandran, Ryan Foley, Kai Zhao, Mike Dusenberry, Yuan Liu, Pulkit Mehta, Ragha Kotikalapudi, Cha-

lence Safranek-Shrader, Andrew Goodman, Joshua Kessinger, Eran Globen, Prateek Kolhar, Chris Gorgolewski, Ali Ibrahim, Yang Song, Ali Eichenbaum, Thomas Brovelli, Sahitya Potluri, Preethi Lahoti, Cip Baetu, Ali Ghorbani, Charles Chen, Andy Crawford, Shalini Pal, Mukund Sridhar, Petru Gurita, Asier Mujika, Igor Petrovski, Pierre-Louis Cedoz, Chenmei Li, Shiyuan Chen, Niccolò Dal Santo, Siddharth Goyal, Jitesh Punjabi, Karthik Kappaganthu, Chester Kwak, Pallavi LV, Sarmishta Velury, Himadri Choudhury, Jamie Hall, Premal Shah, Ricardo Figueira, Matt Thomas, Minjie Lu, Ting Zhou, Chintu Kumar, Thomas Jurdi, Sharat Chikkerur, Yenai Ma, Adams Yu, Soo Kwak, Victor Ähdel, Sujeevan Rajayogam, Travis Choma, Fei Liu, Aditya Barua, Colin Ji, Ji Ho Park, Vincent Hellendoorn, Alex Bailey, Taylan Bilal, Huanjie Zhou, Mehrdad Khatir, Charles Sutton, Wojciech Rzadkowski, Fiona Macintosh, Roopali Vij, Konstantin Shagin, Paul Medina, Chen Liang, Jinjing Zhou, Pararth Shah, Yingying Bi, Attila Dankovics, Shipra Banga, Sabine Lehmann, Marissa Bredesen, Zifan Lin, John Eric Hoffmann, Jonathan Lai, Raynald Chung, Kai Yang, Nihal Balani, Arthur Bražinskas, Andrei Sozanschi, Matthew Hayes, Héctor Fernández Alcalde, Peter Makarov, Will Chen, Antonio Stella, Liselotte Snijders, Michael Mandl, Ante Kärrman, Paweł Nowak, Xinyi Wu, Alex Dyck, Krishnan Vaidyanathan, Raghavender R, Jessica Mallet, Mitch Rudominer, Eric Johnston, Sushil Mittal, Akhil Udathu, Janara Christensen, Vishal Verma, Zach Irving, Andreas Santucci, Gamaleldin Elsayed, Elnaz Davoodi, Marin Georgiev, Ian Tenney, Nan Hua, Geoffrey Cideron, Edouard Leurent, Mahmoud Alnahlawi, Ionut Georgescu, Nan Wei, Ivy Zheng, Dylan Scandinaro, Heinrich Jiang, Jasper Snoek, Mukund Sundararajan, Xuezhi Wang, Zack Ontiveros, Itay Karo, Jeremy Cole, Vinu Rajashekhar, Lara Tumeh, Eyal Ben-David, Rishub Jain, Jonathan Uesato, Romina Datta, Oskar Bunyan, Shimu Wu, John Zhang, Piotr Stanczyk, Ye Zhang, David Steiner, Subhajit Naskar, Michael Azzam, Matthew Johnson, Adam Paszke, Chung-Cheng Chiu, Jaume Sanchez Elias, Afroz Mohiuddin, Faizan Muhammad, Jin Miao, Andrew Lee, Nino Vieillard, Jane Park, Jiageng Zhang, Jeff Stanway, Drew Garmon, Abhijit Karmarkar, Zhe Dong, Jong Lee, Aviral Kumar, Luowei Zhou, Jonathan Evens, William Isaac, Geoffrey Irving, Edward Loper, Michael Fink, Isha Arkatkar, Nanxin Chen, Izhak Shafran, Ivan Petrychenko, Zhe Chen, Johnson Jia, Anselm Levskaya, Zhenkai Zhu, Peter Grabowski, Yu Mao, Alberto Magni, Kaisheng Yao, Javier Snaider, Norman Casagrande, Evan Palmer, Paul Suganthan, Alfonso Castaño, Irene Giannoumis, Wooyeol Kim, Mikołaj Rybiński, Ashwin Sreevatsa, Jennifer Prendki, David Soergel, Adrian Goedeckemeyer, Willi Gierke, Mohsen Jafari, Meenu Gaba, Jeremy

Wiesner, Diana Gage Wright, Yawen Wei, Harsha Vashisht, Yana Kulizhskaya, Jay Hoover, Maigo Le, Lu Li, Chimezie Iwuanyanwu, Lu Liu, Kevin Ramirez, Andrey Khorlin, Albert Cui, Tian LIN, Marcus Wu, Ricardo Aguilar, Keith Pallo, Abhishek Chakladar, Ginger Perng, Elena Allica Abellan, Mingyang Zhang, Ishita Dasgupta, Nate Kushman, Ivo Penchev, Alena Repina, Xihui Wu, Tom van der Weide, Priya Ponnapalli, Caroline Kaplan, Jiri Simsa, Shuangfeng Li, Olivier Dousse, Fan Yang, Jeff Piper, Nathan Ie, Rama Pasumarthi, Nathan Lintz, Anitha Vijayakumar, Daniel Andor, Pedro Valenzuela, Minnie Lui, Cosmin Paduraru, Daiyi Peng, Katherine Lee, Shuyuan Zhang, Somer Greene, Duc Dung Nguyen, Paula Kurylowicz, Cassidy Hardin, Lucas Dixon, Lili Janzer, Kiam Choo, Ziqiang Feng, Biao Zhang, Achintya Singhal, Dayou Du, Dan McKinnon, Natasha Antropova, Tolga Bolukbasi, Orgad Keller, David Reid, Daniel Finchelstein, Maria Abi Raad, Remi Crocker, Peter Hawkins, Robert Dadashi, Colin Gaffney, Ken Franko, Anna Bulanova, Rémi Leblond, Shirley Chung, Harry Askham, Luis C. Cobo, Kelvin Xu, Felix Fischer, Jun Xu, Christina Sorokin, Chris Alberti, Chu-Cheng Lin, Colin Evans, Alek Dimitriev, Hannah Forbes, Dylan Banarse, Zora Tung, Mark Omernick, Colton Bishop, Rachel Sterneck, Rohan Jain, Jiawei Xia, Ehsan Amid, Francesco Piccinno, Xingyu Wang, Praseem Banzal, Daniel J. Mankowitz, Alex Polozov, Victoria Krakovna, Sasha Brown, MohammadHossein Bateni, Dennis Duan, Vlad Firoiu, Meghana Thotakuri, Tom Natan, Matthieu Geist, Ser tan Girgin, Hui Li, Jiayu Ye, Ofir Roval, Reiko Tojo, Michael Kwong, James Lee-Thorp, Christopher Yew, Danila Sinopalnikov, Sabela Ramos, John Mellor, Abhishek Sharma, Kathy Wu, David Miller, Nicolas Sonnerat, Denis Vnukov, Rory Greig, Jennifer Beattie, Emily Caveness, Libin Bai, Julian Eisenschlos, Alex Korchemniy, Tomy Tsai, Mimi Jasarevic, Weize Kong, Phuong Dao, Zeyu Zheng, Frederick Liu, Fan Yang, Rui Zhu, Tian Huey Teh, Jason Sanmiya, Evgeny Gladchenko, Nejc Trdin, Daniel Toyama, Evan Rosen, Sasan Tavakkol, Linting Xue, Chen Elkind, Oliver Woodman, John Carpenter, George Papamakarios, Rupert Kemp, Sushant Kafle, Tanya Grunina, Rishika Sinha, Alice Talbert, Diane Wu, Denese Owusu-Afriyie, Cosmo Du, Chloe Thornton, Jordi Pont-Tuset, Pradyumna Narayana, Jing Li, Saaber Fatehi, John Wieting, Omar Ajmeri, Benigno Uria, Yeongil Ko, Laura Knight, Amélie Héliou, Ning Niu, Shane Gu, Chenxi Pang, Yeqing Li, Nir Levine, Ariel Stolovich, Rebeca Santamaria-Fernandez, Sonam Goenka, Wenny Yustalim, Robin Strudel, Ali Elqursh, Charlie Deck, Hyo Lee, Zonglin Li, Kyle Levin, Raphael Hoffmann, Dan Holtmann-Rice, Olivier Bachem, Sho Arora, Christy Koh, Soheil Hassas Yeganeh, Siim Põder, Mukarram Tariq, Yanhua Sun, Lucian Ionita, Mojtaba

Seyedhosseini, Pouya Tafti, Zhiyu Liu, Anmol Gulati, Jasmine Liu, Xinyu Ye, Bart Chrzaszcz, Lily Wang, Nikhil Sethi, Tianrun Li, Ben Brown, Shreya Singh, Wei Fan, Aaron Parisi, Joe Stanton, Vinod Koverkathu, Christopher A. Choquette-Choo, Yunjie Li, TJ Lu, Abe Ittycheriah, Prakash Shroff, Mani Varadarajan, Sanaz Bahargam, Rob Willoughby, David Gaddy, Guillaume Desjardins, Marco Cornero, Brona Robenek, Bhavishya Mittal, Ben Albrecht, Ashish Shenoy, Fedor Moiseev, Henrik Jacobsson, Alireza Ghaffarkhah, Morgane Rivière, Alanna Walton, Clément Crepy, Alicia Parrish, Zongwei Zhou, Clement Farabet, Carey Radebaugh, Praveen Srinivasan, Claudia van der Salm, Andreas Fidjeland, Salvatore Scellato, Eri Latorre-Chimoto, Hanna Klimczak-Plucińska, David Bridson, Dario de Cesare, Tom Hudson, Piermaria Mendolicchio, Lexi Walker, Alex Morris, Matthew Mauger, Alexey Guseynov, Alison Reid, Seth Odoom, Lucia Loher, Victor Cotruta, Madhavi Yenugula, Dominik Grewe, Anastasia Petrushkina, Tom Duerig, Antonio Sanchez, Steve Yadlowsky, Amy Shen, Amir Globerson, Lynette Webb, Sahil Dua, Dong Li, Surya Bhupatiraju, Dan Hurt, Haroon Qureshi, Ananth Agarwal, Tomer Shani, Matan Eyal, Anuj Khare, Shreyas Rammohan Belle, Lei Wang, Chetan Tekur, Mihir Sanjay Kale, Jinliang Wei, Ruoxin Sang, Brennan Saeta, Tyler Liechty, Yi Sun, Yao Zhao, Stephan Lee, Pandu Nayak, Doug Fritz, Manish Reddy Vuyyuru, John Aslanides, Nidhi Vyas, Martin Wicke, Xiao Ma, Evgenii Eltyshev, Nina Martin, Hardie Cate, James Manyika, Keyvan Amiri, Yelin Kim, Xi Xiong, Kai Kang, Florian Luisier, Nilesh Tripuraneni, David Madras, Mandy Guo, Austin Waters, Oliver Wang, Joshua Ainslie, Jason Baldridge, Han Zhang, Garima Pruthi, Jakob Bauer, Feng Yang, Riham Mansour, Jason Gelman, Yang Xu, George Polovets, Ji Liu, Honglong Cai, Warren Chen, XiangHai Sheng, Emily Xue, Sherjil Ozair, Christof Angermueller, Xiaowei Li, Anoop Sinha, Weiren Wang, Julia Wiesinger, Emmanouil Koukoumidis, Yuan Tian, Anand Iyer, Madhu Gurumurthy, Mark Goldenson, Parashar Shah, MK Blake, Hongkun Yu, Anthony Urbanowicz, Jennimaria Palomaki, Chrisantha Fernando, Ken Durden, Harsh Mehta, Nikola Momchev, Elahe Rahimtoroghi, Maria Georgaki, Amit Raul, Sebastian Ruder, Morgan Redshaw, Jinhyuk Lee, Denny Zhou, Komal Jalan, Dinghua Li, Blake Hechtman, Parker Schuh, Milad Nasr, Kieran Milan, Vladimir Mikulik, Juliana Franco, Tim Green, Nam Nguyen, Joe Kelley, Aroma Mahendru, Andrea Hu, Joshua Howland, Ben Vargas, Jeffrey Hui, Kshitij Bansal, Vikram Rao, Rakesh Ghiya, Emma Wang, Ke Ye, Jean Michel Sarr, Melanie Moranski Preston, Madeleine Elish, Steve Li, Aakash Kaku, Jigar Gupta, Ice Pasupat, Da-Cheng Juan, Milan Someswar, Tejvi M., Xinyun Chen, Aida Amini,

Alex Fabrikant, Eric Chu, Xuanyi Dong, Amruta Muthal, Senaka Buthpitiya, Sarthak Jauhari, Nan Hua, Urvashi Khandelwal, Ayal Hitron, Jie Ren, Larissa Rinaldi, Shahar Drath, Avigail Dabush, Nan-Jiang Jiang, Harshal Godhia, Uli Sachs, Anthony Chen, Yicheng Fan, Hagai Taitelbaum, Hila Noga, Zhuyun Dai, James Wang, Chen Liang, Jenny Hamer, Chun-Sung Ferng, Chenel Elkind, Aviel Atias, Paulina Lee, Vít Listík, Mathias Carlen, Jan van de Kerkhof, Marcin Pikus, Krunoslav Zaher, Paul Müller, Sasha Zykova, Richard Stefanec, Vitaly Gatsko, Christoph Hirnschall, Ashwin Sethi, Xingyu Federico Xu, Chetan Ahuja, Beth Tsai, Anca Stefanoiu, Bo Feng, Keshav Dhandhania, Manish Katyal, Akshay Gupta, Atharva Parulekar, Divya Pitta, Jing Zhao, Vivaan Bhatia, Yashodha Bhavnani, Omar Alhadlaq, Xiaolin Li, Peter Danenberg, Dennis Tu, Alex Pine, Vera Filippova, Abhipso Ghosh, Ben Limonchik, Bhargava Urala, Chaitanya Krishna Lanka, Derik Clive, Yi Sun, Edward Li, Hao Wu, Kevin Hongtongsak, Ianna Li, Kalind Thakkar, Kuanysh Omarov, Kushal Majmundar, Michael Alverson, Michael Kucharski, Mohak Patel, Mudit Jain, Maksim Zabelin, Paolo Pelagatti, Rohan Kohli, Saurabh Kumar, Joseph Kim, Swetha Sankar, Vineet Shah, Lakshmi Ramachandruni, Xiangkai Zeng, Ben Bariach, Laura Weidinger, Tu Vu, Alek Andreev, Antoine He, Kevin Hui, Sheleem Kashem, Amar Subramanya, Sissie Hsiao, Demis Hassabis, Koray Kavukcuoglu, Adam Sadovsky, Quoc Le, Trevor Strohman, Yonghui Wu, Slav Petrov, Jeffrey Dean, and Oriol Vinyals. 2025. Gemini: A family of highly capable multimodal models.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, Katie Millican, et al. 2023. Gemini: a family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.

Gemini Team, Petko Georgiev, Ving Ian Lei, Ryan Burnell, Libin Bai, Anmol Gulati, Garrett Tanzer, Damien Vincent, Zhufeng Pan, Shibo Wang, et al. 2024. Gemini 1.5: Unlocking multimodal understanding across millions of tokens of context. *arXiv preprint arXiv:2403.05530*.

Tristan Thrush, Ryan Jiang, Max Bartolo, Amanpreet Singh, Adina Williams, Douwe Kiela, and Candace Ross. 2022a. Winoground: Probing vision and language models for visio-linguistic compositionality. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5238–5248.

Tristan Thrush, Ryan Jiang, Max Bartolo, Amanpreet Singh, Adina Williams, Douwe Kiela, and Candace Ross. 2022b. Winoground: Probing vision and language models for visio-linguistic compositionality. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5238–5248.

William Timkey and Marten van Schijndel. 2021. All bark and no bite: Rogue dimensions in transformer language models obscure representational quality. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4527–4546, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Annika Tjuka. 2021. A list of color, emotion, and human body part concepts.

Michael Toker, Hadas Orgad, Mor Ventura, Dana Arad, and Yonatan Belinkov. 2024. Diffusion lens: Interpreting text encoders in text-to-image pipelines. *arXiv preprint arXiv:2403.05846*.

Suramya Tomar. 2006. Converting video formats with ffmpeg. *Linux Journal*, 2006(146):10.

Michael Tomasello. 2010. *Origins of human communication*. MIT press.

Shengbang Tong, David Fan, Jiachen Li, Yunyang Xiong, Xinlei Chen, Koustuv Sinha, Michael Rabbat, Yann LeCun, Saining Xie, and Zhuang Liu. 2025. Metamorph: Multimodal understanding and generation via instruction tuning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 17001–17012.

Shengbang Tong, David Fan, John Nguyen, Ellis Brown, Gaoyue Zhou, Shengyi Qian, Boyang Zheng, Théophane Vallaeys, Junlin Han, Rob Fergus, Naila Murray, Marjan Ghazvininejad, Mike Lewis, Nicolas Ballas, Amir Bar, Michael Rabbat, Jakob Verbeek, Luke Zettlemoyer, Koustuv Sinha, Yann LeCun, and Saining Xie. 2026. Beyond language modeling: An exploration of multimodal pretraining. *arXiv preprint arXiv:2603.03276*.

Shengbang Tong, Zhuang Liu, Yuexiang Zhai, Yi Ma, Yann LeCun, and Saining Xie. 2024. Eyes wide shut? exploring the visual shortcomings of multimodal llms. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9568–9578.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models.

Maria Tsimpoukelli, Jacob L Menick, Serkan Cabi, SM Eslami, Oriol Vinyals, and Felix Hill. 2021. Multimodal few-shot learning with frozen language models. *Advances in Neural Information Processing Systems*, 34:200–212.

Ada Defne Tür, Gaurav Kamath, Joyce Chai, Siva Reddy, and Benno Krojer. 2026. Would you still call this dax? novel visual references in vlms and humans.

Ramakrishna Vedantam, Samy Bengio, Kevin Murphy, Devi Parikh, and Gal Chechik. 2017. Context-Aware Captions from Context-Agnostic Supervision. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1070–1079, Honolulu, HI. IEEE.

Constantin Venhoff, Ashkan Khakzar, Sonia Joseph, Philip Torr, and Neel Nanda. 2025. How visual representations map to language feature space in multimodal llms. *arXiv preprint arXiv:2506.11976*.

Tomás Vergara-Browne, Darshan Patil, Ivan Titov, Siva Reddy, Tiago Pimentel, and Marius Mosbach. 2026. Operationalising the superficial alignment hypothesis via task complexity. In *Proceedings of the 43rd International Conference on Machine Learning (ICML)*.

Elena Voita, Rico Sennrich, and Ivan Titov. 2019. The bottom-up evolution of representations in the transformer: A study with machine translation and

language modeling objectives. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4396–4406, Hong Kong, China. Association for Computational Linguistics.

Harm de Vries, Florian Strub, Sarath Chandar, Olivier Pietquin, Hugo Larochelle, and Aaron C. Courville. 2017. Guesswhat?! visual object discovery through multi-modal dialogue. In *Conference on Computer Vision and Pattern Recognition (CVPR)*.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. 2024a. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. 2024b. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*.

T. Wang, K. Lin, L. Li, C. Lin, Z. Yang, H. Zhang, Z. Liu, and L. Wang. 2023a. Equivariant similarity for vision-language foundation models. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 11964–11974, Los Alamitos, CA, USA. IEEE Computer Society.

Tan Wang, Kevin Lin, Linjie Li, Chung-Ching Lin, Zhengyuan Yang, Hanwang Zhang, Zicheng Liu, and Lijuan Wang. 2023b. Equivariant similarity for vision-language foundation models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11998–12008.

Xinlong Wang, Xiaosong Zhang, Zhengxiong Luo, Quan Sun, Yufeng Cui, Jinsheng Wang, Fan Zhang, Yueze Wang, Zhen Li, Qiying Yu, Yingli Zhao, Yulong Ao, Xuebin Min, Tao Li, Boya Wu, Bo Zhao, Bowen Zhang, Liangdong Wang,

Guang Liu, Zheqi He, Xi Yang, Jingjing Liu, Yonghua Lin, Tiejun Huang, and Zhongyuan Wang. 2024c. Emu3: Next-token prediction is all you need.

Peter West, Ximing Lu, Nouha Dziri, Faeze Brahman, Linjie Li, Jena D. Hwang, Liwei Jiang, Jillian Fisher, Abhilasha Ravichander, Khyathi Chandu, Benjamin Newman, Pang Wei Koh, Allyson Ettinger, and Yejin Choi. 2024. The generative AI paradox: "what it can create, it may not understand". In *The Twelfth International Conference on Learning Representations*.

Gregor Wiedemann, Steffen Remus, Avi Chawla, and Chris Biemann. 2019. Does BERT make any sense? interpretable word sense disambiguation with contextualized embeddings. In *Proceedings of the 15th Conference on Natural Language Processing, KONVENS 2019, Erlangen, Germany, October 9-11, 2019*.

Thaddäus Wiedemer, Yuxuan Li, Paul Vicol, Shixiang Shane Gu, Nick Matarese, Kevin Swersky, Been Kim, Priyank Jaini, and Robert Geirhos. 2025. Video models are zero-shot learners and reasoners. *arXiv preprint arXiv:2509.20328*.

Sarah Wiegreffe and Yuval Pinter. 2019. Attention is not not explanation. In *Proceedings of the 2019 conference on empirical methods in natural language processing and the 9th international joint conference on natural language processing (EMNLP-IJCNLP)*, pages 11–20.

Deirdre Wilson and Dan Sperber. 1998. Pragmatics and time. *Pragmatics and Beyond New Series*, pages 1–22.

Penghao Wu and Saining Xie. 2023. V*: Guided visual search as a core mechanism in multimodal llms. *arXiv preprint arXiv:2312.14135*.

Zhaofeng Wu, Xinyan Velocity Yu, Dani Yogatama, Jiasen Lu, and Yoon Kim. 2025. The semantic hub hypothesis: Language models share semantic representations across languages and modalities. In *The Thirteenth International Conference on Learning Representations*.

Jiannan Xiang, Guangyi Liu, Yi Gu, Qiyue Gao, Yuting Ning, Yuheng Zha, Zeyu Feng, Tianhua Tao, Shibo Hao, Yemin Shi, Zhengzhong Liu, Eric P. Xing, and Zhiting Hu. 2024. Pandora: Towards general world model with natural language actions and video states. `https://arxiv.org/abs/2406.09698`.

Ning Xie, Farley Lai, Derek Doran, and Asim Kadav. 2019. Visual entailment: A novel task for fine-grained image understanding. *arXiv preprint arXiv:1901.06706*.

Shaoan Xie, Zhifei Zhang, Zhe Lin, Tobias Hinz, and Kun Zhang. 2023. Smartbrush: Text and shape guided object inpainting with diffusion model. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 22428–22437.

Jun Xu, Tao Mei, Ting Yao, and Yong Rui. 2016. Msr-vtt: A large video description dataset for bridging video and language. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5288–5296.

An Yan, Xin Wang, Tsu-Jui Fu, and William Yang Wang. 2021. L2C: Describing visual differences needs semantic understanding of individuals. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 2315–2320, Online. Association for Computational Linguistics.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, Jin Xu, Jingren Zhou, Jinze Bai, Jinzheng He, Junyang Lin, Kai Dang, Keming Lu, Ke-Yang Chen, Kexin Yang, Mei Li, Min Xue, Na Ni, Pei Zhang, Peng Wang, Ru Peng, Rui Men, Ruize Gao, Runji Lin, Shijie Wang, Shuai Bai, Sinan Tan, Tianhang Zhu, Tianhao Li, Tianyu Liu, Wenbin Ge, Xiaodong Deng, Xiaohuan Zhou, Xingzhang Ren, Xinyu Zhang, Xipin Wei, Xuancheng Ren, Yang Fan, Yang Yao, Yichang Zhang, Yunyang Wan, Yunfei Chu, Zeyu Cui, Zhenru Zhang, and Zhi-Wei Fan. 2024a. Qwen2 technical report. *ArXiv*, abs/2407.10671.

Sherry Yang, Yilun Du, Seyed Kamyar Seyed Ghasemipour, Jonathan Tompson, Leslie Pack Kaelbling, Dale Schuurmans, and Pieter Abbeel. 2024b. Learning interactive real-world simulators. In *The Twelfth International Conference on Learning Representations*.

Qinghao Ye, Xianhan Zeng, Fu Li, Chunyuan Li, and Haoqi Fan. 2025. Painting with words: Elevating detailed image captioning with benchmark and alignment learning. In *The Thirteenth International Conference on Learning Representations*.

Ahmet Burak Yildirim, Vedat Baday, Erkut Erdem, Aykut Erdem, and Aysegul Dundar. 2023. Inst-inpaint: Instructing to remove objects with diffusion models. *arXiv preprint arXiv:2304.03246*.

Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. 2014. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2:67–78. Place: Cambridge, MA Publisher: MIT Press.

Licheng Yu, Patrick Poirson, Shan Yang, Alexander C. Berg, and Tamara L. Berg. 2016. Modeling context in referring expressions. *ArXiv*, abs/1608.00272.

Qihang Yu, Mark Weber, Xueqing Deng, Xiaohui Shen, Daniel Cremers, and Liang-Chieh Chen. 2024. An image is worth 32 tokens for reconstruction and generation. *Advances in Neural Information Processing Systems*, 37:128940–128966.

Mert Yuksekgonul, Federico Bianchi, Pratyusha Kalluri, Dan Jurafsky, and James Zou. 2023a. When and why vision-language models behave like bags-of-words, and what to do about it? In *The Eleventh International Conference on Learning Representations*.

Mert Yuksekgonul, Federico Bianchi, Pratyusha Kalluri, Dan Jurafsky, and James Zou. 2023b. When and why vision-language models behave like bags-of-words, and what to do about it? In *The Eleventh International Conference on Learning Representations*.

Rowan Zellers, Jiasen Lu, Ximing Lu, Youngjae Yu, Yanpeng Zhao, Mohammadreza Salehi, Aditya Kusupati, Jack Hessel, Ali Farhadi, and Yejin Choi. 2022. Merlot reserve: Multimodal neural script knowledge through vision and language and sound. In *CVPR*.

Rowan Zellers, Ximing Lu, Jack Hessel, Youngjae Yu, Jae Sung Park, Jize Cao, Ali Farhadi, and Yejin Choi. 2021. Merlot: Multimodal neural script knowledge models. *Advances in Neural Information Processing Systems*, 34:23634–23651.

Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. 2023. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 11975–11986.

Kai Zhang, Lingbo Mo, Wenhu Chen, Huan Sun, and Yu Su. 2024. Magicbrush: A manually annotated dataset for instruction-guided image editing. *Advances in Neural Information Processing Systems*, 36.

Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. 2023a. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3836–3847.

Shu Zhang, Xinyi Yang, Yihao Feng, Can Qin, Chia-Chih Chen, Ning Yu, Zeyuan Chen, Haiquan Wang, Silvio Savarese, Stefano Ermon, Caiming Xiong, and Ran Xu. 2023b. Hive: Harnessing human feedback for instructional visual editing. *ArXiv*, abs/2303.09618.

Zhen Zhang, Xuehai He, Weixiang Yan, Ao Shen, Chenyang Zhao, and Xin Eric Wang. 2026. Soft thinking: Unlocking the reasoning potential of LLMs in continuous concept space. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

Zhi Zhang, Srishti Yadav, Fengze Han, and Ekaterina Shutova. 2025. Cross-modal information flow in multimodal large language models. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 19781–19791.

Xiangyu Zhao, Peiyuan Zhang, Kexian Tang, Xiaorong Zhu, Hao Li, Wenhao Chai, Zicheng Zhang, Renqiu Xia, Guangtao Zhai, Junchi Yan, et al. 2026. Envisioning beyond the pixels: Benchmarking reasoning-informed visual editing. *Advances in Neural Information Processing Systems*, 38.

Boyang Zheng, Nanye Ma, Shengbang Tong, and Saining Xie. 2026. Diffusion transformers with representation autoencoders. In *The Fourteenth International Conference on Learning Representations*.

Chunting Zhou, LILI YU, Arun Babu, Kushal Tirumala, Michihiro Yasunaga, Leonid Shamis, Jacob Kahn, Xuezhe Ma, Luke Zettlemoyer, and Omer Levy. 2025. Transfusion: Predict the next token and diffuse images with one multi-modal model. In *The Thirteenth International Conference on Learning Representations*.

Kankan Zhou, Eason Lai, and Jing Jiang. 2022. VLStereoSet: A study of stereotypical bias in pre-trained vision-language models. In *Proceedings of the 2nd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 12th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 527–538, Online only. Association for Computational Linguistics.

Siyuan Zhou, Yilun Du, Jiaben Chen, Yandong Li, Dit-Yan Yeung, and Chuang Gan. 2024. Robodreamer: Learning compositional world models for robot imagination. *arXiv preprint arXiv:2404.12377*.

Chengxu Zhuang, Evelina Fedorenko, and Jacob Andreas. 2024. Visual grounding helps learn word meanings in low-data regimes. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 1311–1329, Mexico City, Mexico. Association for Computational Linguistics.